

Taking the ASD Exam

Sam Mason
Mobile Intelligence

Published by the Symbian Developer Network

Version: 1.0 – June 2007

1	INTRODUCTION	2
2	DISCOVERY	3
3	PREPARATION	3
4	EXECUTION.....	6
5	REFERENCES	11
6	AUTHOR PROFILE.....	11

1 Introduction

The famous Dutch Computer Scientist Edsger Dijkstra was well known for the statement that “Computer Science is no more about computers than astronomy is about telescopes”. In direct parallel, programming for Symbian OS is less about mobile phone software development specifically than it is about professional engineering under resource constrained conditions. Of the fourteen sections in the ASD syllabus, only about two or three can really be said to be mobile phone specific and even that is arguable. The result is an excellent course in computer science and engineering principles.

The Accredited Symbian Developer (ASD) program is run by Majinate [1] and is designed to ensure that certification can only be achieved through a combination of theory and practice. Apart from trying to ensure that Symbian OS software engineers have a minimum level of competence, one of the main purposes that the program serves is to prevent just anyone from claiming to be a Symbian engineer – as happens with many other industry certifications.

Once you achieve ASD status, you are entitled to use the ASD logo on your business cards, resume and website. You also gain access to the ASD member’s area of the Majinate website where you can keep tabs on what’s going on in the Symbian world. It also forms a prerequisite for the new S60 Accredited Developer program [2] supported by Nokia, so if you’re into certifications this is the one to start with.

I’m going to assume that if you’re reading this that you know the basics of what the ASD is, so I won’t go into excessive detail here. The test is comprised of a series of multiple choice questions presented using adaptive questioning techniques. The test works by first assessing the competency level of the candidate and once it has been determined, it then tests them progressively to push them to their limits until they display a consistent level of performance.

At this point in time, there are 48 questions and you get two hours to complete them in. The pass mark is a closely guarded Symbian secret so you have to do the best that you can. It usually takes about 60-70 minutes and you can sit it in supervised or unsupervised mode.

Doesn’t sound too bad does it? Well here’s the catch – each question has five possible answers of which up to three can be correct. So forget the old high school approach where you thought you had a 20% chance of guessing a correct answer. Marks are deducted for neglecting to select a correct answer, for selecting an incorrect answer and for taking too long to answer at all. Some answers are more “right” or “wrong” than others and are therefore weighted accordingly. You cannot go back to previous questions and change your answers either – you get one shot at each question and that’s it.

One phrase that I heard summed it up very nicely – “it is not a test designed to make you feel good about yourself”. It is not easy, for it is a very challenging examination. That may sound daunting, but therein lies the value of the certification.

Over the next few years we’re going to see an explosive growth in mobile phone software development and what will follow is a large number of poorly skilled people claiming to be Symbian programmers. This is what the ASD is designed to prevent. If it was easy, everyone would have one (like Microsoft’s MCAD).

The purpose of this paper is to outline my experiences in becoming an ASD. I will discuss what I did to prepare for it, my expectations versus reality, results, benefits and I’ll conclude with some advice on getting through it all.

I hope that this will both encourage you to become an ASD and help you on your way.

Good luck!

2 Discovery

I first heard about the ASD program at the Nokia conference in November 2006 held in Sydney, Australia. At that time I had been working on various versions of Symbian OS for a little over two years. I had not however, had anything to do with version 9.x, and to be honest, I was a little spooked about these “capabilities”. I was also hearing about “Platsec”, “data caging” and “Symbian Signed” – and a number of other phrases that didn’t make a whole lot of sense.

Like most techies, I spend much of my time learning new technologies and techniques. And also like all (ok well most) techies I have often managed to teach myself a whole lot of bad techniques in the process. I know from my experiences in the past with Java certifications that the gap between what you think you know and what you really know can be a lot wider than first thought – and this is why I’m a big fan of certifications. You are forced to pay attention to detail and if you’re doing that, you’re going to be producing better code in less time. You’re also going to understand *why* you’re using the techniques you need. This translates to making you much more cost effective – which of course your employers are going to love.

Finding out about the ASD was the first step for me. Within a day, I was armed with two Symbian Press books - “The Accredited Symbian Developer Primer” and a copy of “Symbian OS Platform Security” the latter of which was given to me at the conference.

I knew instantly that I would do this even before I finished my new product line. I knew that it would fill in the gaps I didn’t know were even there and I knew that I would be a better programmer for having done it. What I didn’t realise was just how true this would turn out to be.

I’d worked as a computer scientist for 10 years in Java, C++ and C# designing and building back-end systems for businesses around Australia. I’d done a few industry certificates before and I’d just finished a Master’s degree in AI. In addition, I knew that the pass mark was based on testing Symbian engineers with about 6 months experience straight out of university. In short, I was pretty confident.

So I downloaded a set of sample questions from Meme Education [3], sat down with pen, paper and a fresh coffee thinking I’d just do a quick run through and get back to work. Wrong, wrong, wrong.

Five minutes later, I was digging out my battered copy of “The C++ Programming Language” [4] and asking myself – “Um... how *do* I get full value semantics for a user defined class again?”

So I had discovered the ASD, and the ASD had discovered me.

I had some work to do.

3 Preparation

When I set out to do an exam, I always read the available text(s) from cover to cover at first, and then read them again making notes as I go along. Once my notes are complete, I read them whenever I get a chance (on the bus, at the gym or in the pub). By the end, even without really trying that hard, I find that I have completely learned all of the concepts required and suddenly passing the exam is a secondary goal. The primary achievement is a rather large hoard of extremely valuable knowledge.

3.1.1 Text Books

There’s no going past the “Accredited Symbian Primer” by Mark Jacobs and Jo Stichbury, published by Symbian Press. Since they are two of the people involved in the ASD syllabus and

testing process, you can be sure that it covers pretty much everything. Quite apart from that, it also covers off a number of lesser known exceptions that make life interesting for Symbian OS programming. Here's one of my favourite examples of that:

If you're using a `TFixedArray` you can access elements of the array either by calling the `At()` method, or by using the overloaded `[]` operator. However, if you use `At()`, there is the performance penalty of bounds checking using assertions – *and this applies even in release builds!* If you use the `[]` operator, no assertions are used and access is faster as a consequence.

Now, it doesn't really matter whether you get asked that during the exam or not – the point is that you've learned something that you can probably use immediately. The Primer is filled with countless examples of this type. Even if you don't take the test, just read through the Primer. In my opinion, it should be mandatory for all serious Symbian developers.

The other textbook that I will swear by is "Symbian OS Explained" solely authored by Jo Stichbury. This book is already well known to be one of the best Symbian titles. For me the real value here was the detail the author has gone into in describing the client-server framework. You get walked through the complete source of a client and server under Symbian OS. The version in the book is pre version 9, but Jo has provided a version of the code updated for Symbian OS v9 on the Meme website.

Since I have collected most of the Symbian Press titles [5], I also used them as secondary texts if I wanted to follow up something from a different angle (Harrison, Heath and Babin).

3.1.2 Sample Questions

You can download a set of sample questions from the Meme Education website. The current version is 1.1 (updated in March, but I had the older version 1.0) consisting of 48 questions rated for difficulty. Meme Education is run by the same people who wrote the Primer, so the questions are of immense value if you want to get a feel for where your skill level is before you take the test.

Also included are the answers to all 48 questions. What is *not* included is why – there are no explanations. I think that this is great – it forces you to work out why your answer was wrong. Here's the sort of simple thing that can trip you up (adapted from the samples):

Given the code below for `TestFunction()`, which of the following statements are correct?

```
void TestFunction()
{
1   _LIT(KHello, "Hello!");
2   TBufC<6> hello(KHello);
...

```

A. After executing line 2, a call to `hello.MaxLength()` returns 6.
 B. ...

The point here is that constant descriptors do not have a `MaxLength()` method – only modifiable descriptors deriving from `TDes` have `MaxLength()`. Here's another example:

Which of the following statements about Symbian OS capabilities are incorrect?

...

B. The following specification in an MMP file grants privilege to access the user's files stored anywhere on the phone or on removable media:

```
CAPABILITY ReadUserData
```

It looks right – to read user data (i.e. PIM data) you need the ReadUserData capability. In fact it's incorrect. Read it again and remember that you need the AllFiles capability to read files containing user data stored *anywhere* on the phone.

As you can see, questions can be phrased to look like they're testing a particular concept, when in fact they are testing whether you've simply memorised a series of facts without understanding where they fit into the larger concepts in Symbian development.

The take home lesson is that you need to be very careful in this exam. The sample questions give you a chance to see the kind of terminology used and the phraseology of the answers presented. I cannot stress this enough:

READ EACH QUESTION CAREFULLY

3.1.3 Practical

When all is said and done, you have to get your hands dirty if you're going to really understand a technology. Although I had done a lot of Symbian work, I still found it necessary to put together a number of test applications to work through various concepts in detail.

I used the sample code to build a simple server (something which I had not done before) as well as experiment with some of the overloads that you can use in thread and process creation. More importantly, I acquired an N73 during this period. This allowed me to actually work through self signing a SIS file using CreateSIS and deploying it to S60 3rd edition hardware.

Another thing that I did was to make sure that I was comfortable with RBufs, which I hadn't used before. After the event it's straightforward, but working through an example that I invented myself also had the added benefit of making me first think about how I would use an RBuf (design) before actually doing so (implementation).

You can either use a simple console app or a basic GUI app to do your testing in – I used both but it doesn't matter. Just remember that your code is for learning only, so don't get caught up trying to make it efficient and building libraries – you're trying to cement in a concept, so keep it simple and focus on the concepts.

By all means study hard, but leverage your own experience and make sure you get as much practice as possible on anything you're not strong on.

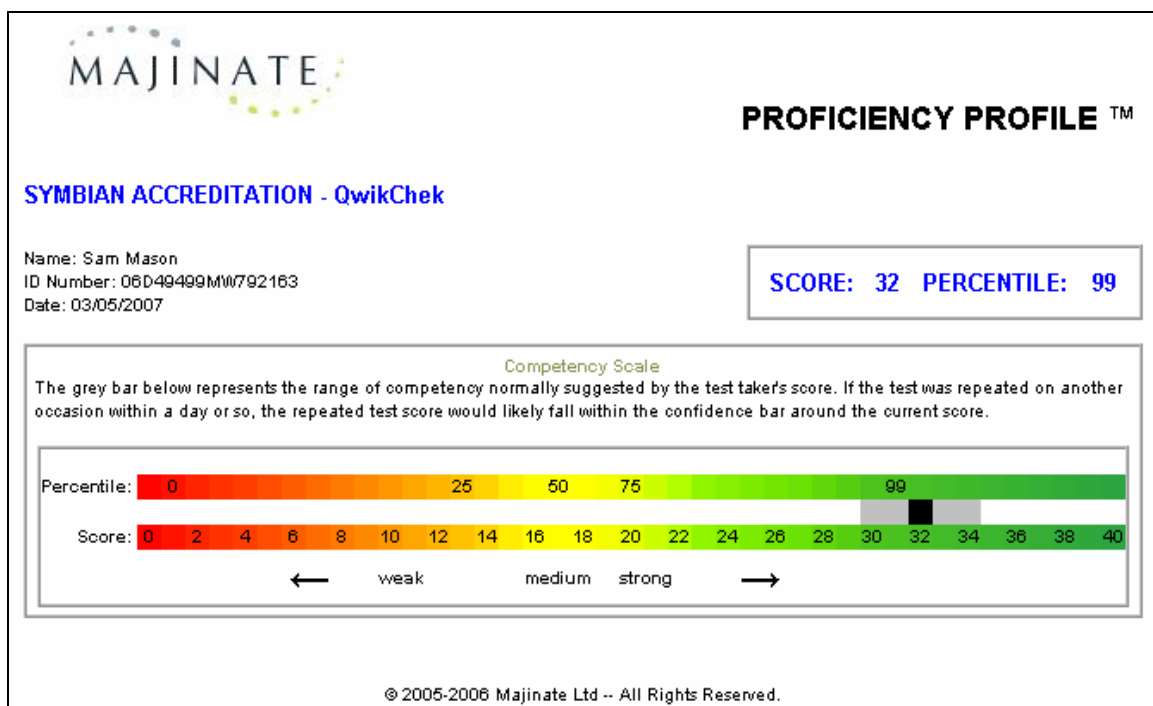
3.1.4 Practice Exam

I also did the ASD practice exam through the Majinate website. Of course you don't have to do this, but I would highly recommend it. It costs £40 and you sit it unsupervised online through your browser. It will terminate the examination immediately if you try and open other applications while you're using it, so be sure to close down everything else beforehand.

This is where you really find out where you're at and whether you're ready to take the test. The questions are exam standard and it is exactly like doing the real test except that you only get 35 minutes to do as many questions as you can.

I was really happy with the practice exam although more than once I clicked for the next question before selecting all the options I wanted – and as I said earlier there is no going back. So it's a good idea to take your time (within reason) on each question. You should aim to be able to answer each question in about 90 seconds.

When the practice exam concludes, you will get emailed your results. These are pretty high level and just give you a measure of your performance relative to others who have taken the test in the past (although this pool of people changes as the syllabus develops over time) as well as an absolute score as seen below:



The important thing here is your percentile rating. This is probably the best indicator of your readiness to take the real test, so leave it until you've covered off everything before you jump in. Depending on what country you're in, £40 can translate to a lot of money, so you will definitely want this experience to work for you.

This was my last step before taking the real exam. I was ready.

4 Execution

4.1.1 Logistics

You can actually book and pay for your exam through the Majinate website. You can sit the exam in unsupervised mode online whenever you wish after it has been setup (you get an email telling you how to do this). On the other hand, you may wish for the supervised mode in which case you're going to need to get to a testing centre.

From the outset, I was always going to opt for the supervised option. It's only my personal opinion, but I know that I value the extra challenge. Others may disagree – I just couldn't stand the thought of the words "unsupervised" written on my certificate. In addition, it forced me to work harder so that I wasn't going to waste anyone's time.

In my case, supervision required a fair bit of extra organization as there is currently nowhere in Australia to sit a supervised ASD exam. Luckily for me, Symbian, Majinate and Meme Education were all happy to cover this special case so I flew from Sydney to Melbourne to sit a supervised test in late March 2007.

In most cases, it's probably going to be easier to do the test online unless you're close to the testing centres in India, the US, Canada or the EU – see the Majinate website for more details.

4.1.2 Examination

This section is necessarily being written after the event, but I must say that no amount of speculation prepared me for the reality of this experience. In short, I passed it – which was the important bit, but it was more demanding than I was expecting.

It was only after the event that my supervisor reminded me that extremely hard questions (and for me, most of the ones after about 40 were surprisingly difficult) are a clear indication that you're doing well. I had forgotten this at the time however, and it did distract me from the task at hand – so be ready for it and don't let it affect your performance.

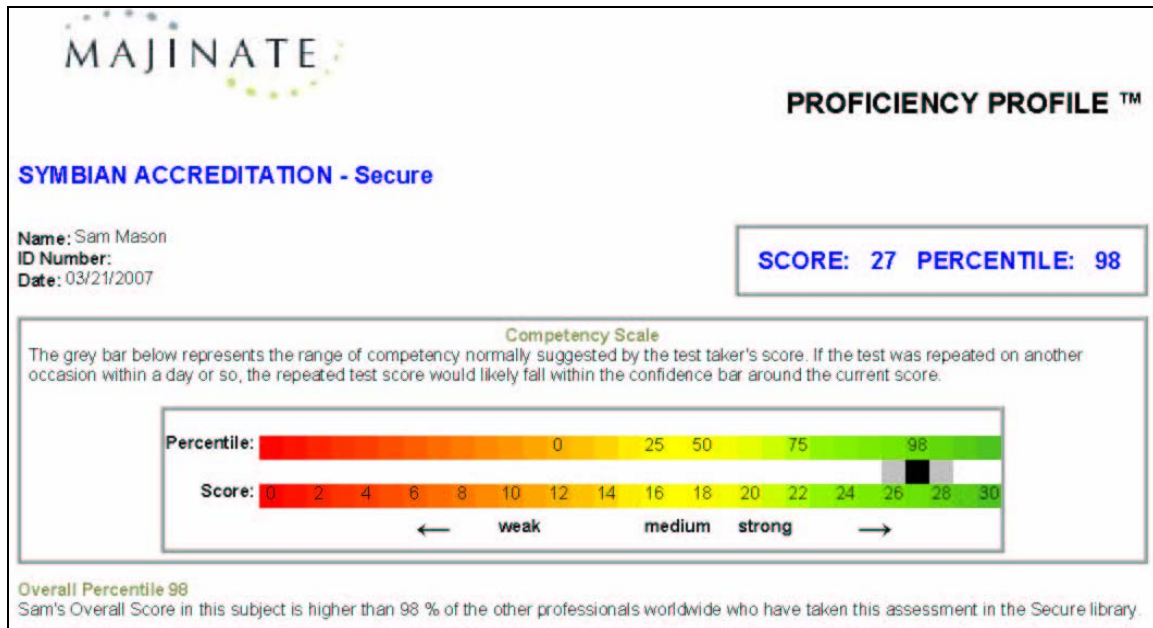
The other thing that surprised me was that I was expecting the exam to start at a relatively easy level and to then get harder from there. In my case, the first question I got was one that I had to guess (not a good start) – however in retrospect I believe that this was more bad luck than anything else.

While I obviously cannot divulge any exam questions specifically, I can say that there was a certain amount of overlap between the set of exam questions that I got and the set of questions from the online sample document and the practice test that I did. However, it's worth remembering that you're getting a small sample set from a large pool of questions, so there are no guarantees. Once again, I strongly recommend that you include the practice questions/exam as essential building blocks of your preparation strategy.

The main difference that I found was that I was expecting the exam to take the same form as the online practice one I did – that is, you keep doing questions until you meet the pass criteria. But it wasn't like that at all. Knowing that you had two hours and 48 questions removed a large amount of the time pressure for me and actually allowed me to proceed quite quickly.

4.1.3 Assessment

After a long lead up, it was over. Within a few minutes, my supervisor was able to log in to determine that I had passed (phew). As before, an email arrived with the assessment results. You have two options for this depending on how you paid for the exam originally – you can elect to have a simple binary result (i.e. – "you passed" or "you failed") for £100 or you can choose to get a detailed breakdown for the cost of £140. I chose the latter and got the following:



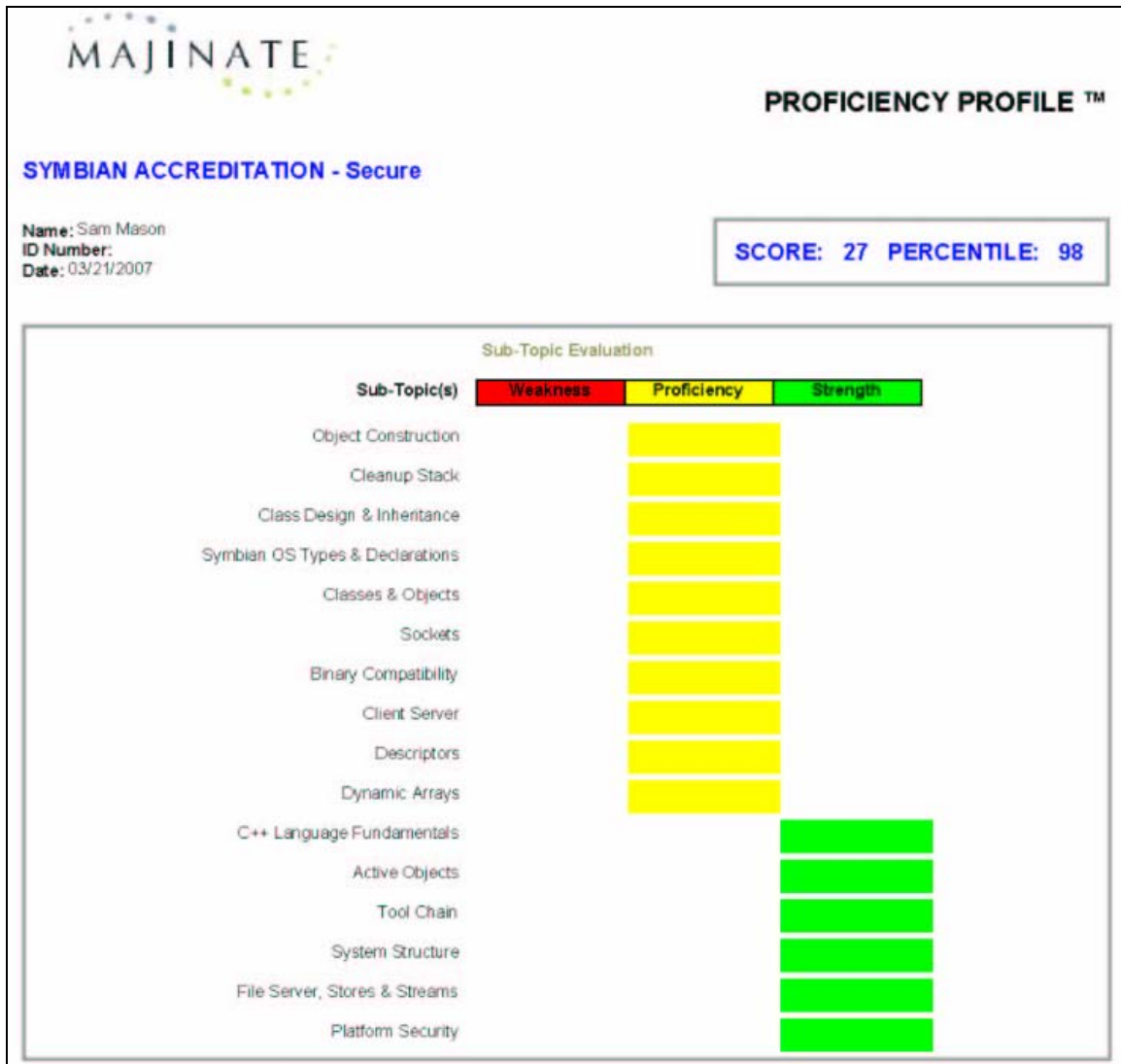
It may be a bit hard to read the last line above, but don't be too impressed by the 98 percentile result. It simply means that my performance was in the top 2% of people who have also done the same exam. It most certainly does not mean that I got 98%!

Below this, (not shown above) you also get a percentile breakdown for your book knowledge, practical experience, conceptual understanding, and work speed. These are particularly useful if you're using this test to assess skills as part of a recruitment process.

The diagram below shows the second part of the results you get. This is a proficiency breakdown over all knowledge categories tested – which pretty much maps one-to-one with the chapters of the ASD Primer.

Personally I found these results surprising as I found out that I was stronger in some areas than I had thought and had obviously answered some questions incorrectly and/or incompletely in areas that I thought I knew very well.

Overall, I was very satisfied with these results – I thought it was a pretty fair reflection of the experience I had and the prep work I had done. More importantly, my final result was within a percentile of that predicted by the practice exam process, therefore validating its relevance and accuracy.



You also get a congratulations email from the people at Majinate supplying your login details for the member's area of the Majinate website.

By the way, if you don't pass the first time, there are no future penalties if you choose to re-take the exam in the future. Your proficiency breakdown can also be used to guide you on what to study to ensure a pass next time.

5 Conclusion

There is no substitute for experience, but in Information Technology, just as in many other fields, there are many things that you either know or you don't. You can't work it out from first principles if you don't know it (at least not in any bounded time under exam conditions). This is why experience must be complemented by theory. To pass the ASD exam you will need both.

5.1.1 Problems

Some of the problems I encountered during this process are subjective and due to my own exposures over the last ten years whereas others were a consequence of the resources available at the time. In particular, I over-prepared and started stumbling over simple pieces of new information that I had learned. This is what happens when your preparation process goes for too long. The best thing to do is get ready, do the practice test and if you think you're ready, do the real thing immediately. This is not an exam that you want to take when your brain is burnt out.

I also had a few problems with the sample exam questions from the Meme Education site as I found that some of them were a little ambiguous and some of the provided answers were incorrect. However this was not a big deal – and the folks at Meme are very approachable about these issues. As of March 2007, these have been addressed in the V1.1 updates however I didn't have these at the time.

In general, the main obstacle I found was simply that I wasn't reading the questions properly and would constantly assume I knew what it was asking. This caused me to get some easy questions wrong when I shouldn't have.

5.1.2 Advice

Hopefully I have now convinced you of this, but just in case, here it is again:

READ EACH QUESTION CAREFULLY

If you are serious about the ASD program and certification, then you are probably going to need to spend some money. You will need to at least purchase the ASD Primer and you should probably do the online practice exam.

Expect it to be difficult because it is. Make sure your experience is augmented by your knowledge of the theory and vice versa – the exam will require both.

In this more fixed structure, time per question is less of an issue, but you are assessed on speed as well, so be prepared on the day so that you can focus.

Try to see the achievement of the certification as secondary to the accrual of detailed knowledge. Try to think of obvious areas that will get tested and try making up your own questions. The more practice you do, the better you are going to be.

5.1.3 Looking Forward

For me, one of the best things that came out of this certification process was that I've re-sharpened my edge. I've had to go back to basics in C++ again, care about critical sections and semaphores and think about what it really means to use a chunk of memory.

I've had to brush up on a number of things that we learned at university – most of which my professional work in Java and C# development had allowed to attenuate over the years. It was great to jump back into it again - there's no doubt I'm a better programmer for the experience.

I whole-heartedly recommend the ASD program for anyone serious about working with Symbian OS. It has been one of the more intense learning processes of my professional life.

I'd like to thank Ian Weston from Majinate, Hamish Willee from Symbian and Mark Jacobs from Meme Education for all their help in making this goal a reality for me. Without them this whole thing just wasn't going to happen.

6 References

- [1] Majinate: <http://www.majinate.com>
- [2] S60 Program: <http://www.majinate.com/curriculumS60.shtml>
- [3] Meme Education: <http://www.meme-education.com>
- [4] Stroustrup, B. *The C++ Programming Language*
- [5] Symbian Press: <http://developer.symbian.com/main/academy/press/index.jsp>

7 Author Profile

Sam Mason is a professional computer scientist based in Sydney Australia. He has architected a number of business systems over the last ten years in such diverse sectors as Human Resources, Supply & Logistics, Entertainment, Procurement and Accounting, Payroll and Finance. Sam holds a number of industry certifications and a Masters degree from UNSW in Autonomous Systems. He is the founding member of Mobile Intelligence (<http://www.mobileintelligence.com.au>), started in 2006, and hopes to specialise exclusively in mobile phone software development moving forward. Sam is particularly interested in using AI techniques to build better applications to make smart phones even smarter

You can contact him at sam@mobileintelligence.com.au.

[Back to Developer Library](#)

Want to be kept informed of new articles being made available on the Symbian Developer Network?

[Subscribe to the Symbian Community Newsletter](#).

The Symbian Community Newsletter brings you, every month, the latest news and resources for Symbian OS.