



64 Tannery Drift
Royston
Herts
UK
SG8 5DE

European Patent Office
Enlarged Board of Appeal
80298 München
Germany

26 April 2009

Case G3/08: President's referral relating to patentability in the field of computing.

I wish to submit this as an amicus curiae brief, answering the questions posed in the referral above. I am inter alia a Chartered IT Professional and a Member of the Institute of Patentees and Inventors.

1. No. An application to patent a computer program simply invalidates itself.
- 2a. No. The explicit mention of essential adjuncts to a computer program cannot make any difference.
- 2b. Yes. The invention must do something new and useful, or do something useful in a new way. What it does and how it does it determines whether it is patentable.
- 3a. Yes, but the answer begs the question what is a technical effect. As well as doing something new and useful, the method used must require an identified inventive step, which cannot simply be the obvious application of prior art or the obvious application of new (or old) discoveries.
- 3b. No. The claimed feature would have to be a specified better form of computer.
- 3c. No. But the question is open to misinterpretation - if the effect is independent of any particular hardware, then the effects achieved are within the present specification of hardware from any supplier, and so can hardly be novel.

If the effect is independent of any particular type of hardware, the answer is even more firmly no, but this time because an existing general purpose machine is simply being realised from a different collection of prior art.

- 4a. Yes. So does the construction of a safe building, so does the connection of domestic gas and water pipes. Technical considerations are necessary but not sufficient for invention.
- 4b. No. Not necessarily - so not 'thus'. If the program contributes to the 'new and useful' nature then it is part of the invention. Whether the patent application should succeed is another matter.
- 4c. No. Irrespective of the answer to 4a. An intermediate technical invention can result in a non-technical patentable improvement (eg. smaller, faster, cheaper to run) to an existing product, but which is dependent on the different technical nature of the program.

Amplifying my views on Q4, if an inventor works out a more efficient algorithm for sorting values, as a logical and mathematical achievement it is not patentable.

Using this new algorithm to produce a faster melon sorter would be a patentable invention - unless (as is likely) it was deemed to be an obvious exploitation of the un-patentable mathematical and logical process. But it is the improved melon sorter that may be patentable, not the program. If, as is likely, a veritable deluge of such patents is the foreseeable consequence of the new algorithm, then there is little doubt that these applications of the new algorithm are obvious.

The grant of a patent is in return for publishing design details sufficient to allow one practised in the

art to realise the invention. In the context of this discussion, this would mean full and detailed disclosure of the programming algorithms such that a competent programmer could realise the invention using a computer system with a completely different set of basic facilities. Emphasising this fact might well deter many 'closed source' software programmers from applying for patent protection, and could invalidate some patents already granted.

On question 3:

The 'by design' general purpose nature of mathematics and logic mean that inventions based on their exploitation are unlikely to result in a patentable invention for this very reason of obviousness if they merely result in a new way of achieving an existing useful product. To stand a chance of being part of a patentable invention they would have to result in something new, useful and not obvious.

The 'by design' general purpose nature of computing apparatus means that inventions based on mathematical and logical processes are most unlikely to be part of a patentable invention in this field since they merely bring the obvious benefits of the algorithm to an existing general purpose product. To be patentable they would probably have to result in a novel form of computer.

On question 2:

The purpose of a patent is to allow society to benefit from the invention, and to reward the inventor for that benefit. The purpose is not to deny society the benefit of an invention which potentially undermines an existing lucrative arrangement. Exploitation should be a requirement for the valid assignment of patent rights, or the grant of an exclusive licence. Patents which are not exploited in products brought to market and freely available within a reasonable time, should be forfeited, on the basis that the invention has failed the 'useful' test. In the context of my answer to 2b above, 'potentially useful' is not enough to ensure the granting of a robust patent.

On question 1:

A computer program is an ordered collection of instructions. These instructions are prior art, intended to facilitate the achievement of general purpose machines. However technically well crafted the assembly and sequencing of these instructions may be, the end result is within the intention of the designer of the instruction set. It is no more patentable than the well crafted assembly and sequencing of the words of a technical reference book or an intricately logical detective novel, and no more patentable than the skilled and technically accurate assembly and sequencing of mathematical symbols in the formal solution of a mathematical problem.



C W Cowsley MA PhD C.Eng CITP MBCS MInst.PI