

## 1 2008 Paper 3 Question 3

A hardware engineer stores a FIFO queue of bits in an int on a platform with 32-bit ints and 8-bit chars using the following C++ class:



<https://www.cl.cam.ac.uk/teaching/exams/pastpapers/y2008p3q3.pdf>

```
class BitQueue{
    int valid_bits; // the number of valid bits held in the queue
    int queue;      // least significant bit is most recent bit added
public:
    BitQueue(): valid_bits(0), queue(0) {}
    void push(int val, int bsize);
    int pop(int bsize);
    int size();
}
```

- (a) Write an implementation of `BitQueue::size`, which should return the number of bits currently held in queue.

```
int size(){
    return valid_bits;
}
```

- (b) Write an implementation of `BitQueue::push`, which places the `bsize` least significant bits from `val` onto `queue` and updates `valid_bits`. An exception should be thrown in cases where data would otherwise be lost.

```
void push(int val, int bsize){
    int j = 1;
    for (int i = 0; i < bsize; i++){
        queue |= (j & val) << valid_bits;
        j *= 2;
    }
    valid_bits += bsize;
}
```

- (c) Write an implementation of `BitQueue::pop`, which takes `bsize` bits from `queue`, provides them as the `bsize` least significant bits in the return value, and updates `valid_bits`. An exception should be thrown when any requested data is unavailable.

```
int pop(int bsize){
    int ret = 0;
    int j = 1;
    for (int i = 0; i < bsize; i++){
        ret |= (j & queue);
        j *= 2;
    }
    queue >>= bsize;
    valid_bits -= bsize;
}
```

- (d) The hardware engineer has built a communication device together with a C++ library function `send` to transmit data with the following declaration.



```
void send(char);
```

Use the BitQueue class to write a C++ definition for:

```
void sendmst(const char* msg);
```

Each of the characters in `msg` should be encoded, in index order, using the following binary codes: 'a'=0, 'b'=10, 'c'=1100 and 'd'=1101. All other characters should be ignored. Successive binary codes should be bit-packed together and the code 111 should be used to denote the end of the message. Chunks of 8-bits should be sent using the `send` function and any remaining bits at the end of a message should be padded with zeros. For example, executing `sendmst("abcd")` should call the `send` function twice, with the binary values 01011001 followed by 10111100.

## 2 2009 Paper 3 Question 1

Explain all of the following C or C++ features. You may use a short fragment of code to complement your explanation.

- (a) The declaration of a C++ class illustrating constructor, variable and method.
- (b) The use of a virtual destructor
- (c) The difference between `malloc()` and `free()`; and `new` and `delete`
- (d) Overloading an operator
- (e) Pointer arithmetic
- (f) Catching and throwing exceptions including the passing of a user-defined structure
- (g) The meaning of the keywords `static` and `const`



<https://www.cl.cam.ac.uk/teaching/exams/pastpapers/y2009p3q1.pdf>

## 3 2012 Paper 3 Question 3

In this question, where appropriate, you may use a short fragment of code to complement your explanation.



<https://www.cl.cam.ac.uk/teaching/exams/pastpapers/y2012p3q3.pdf>

- (alph\*) (i) What is the difference between a local and global variable in C? (Consider variable scope, storage and initialisation)
- (ii) What are the properties of a static member variable in a C++ class?
- (alph\*) (i) Briefly explain pointer arithmetic in C. Give an example code snippet involving pointers in which it would be *inappropriate* to use pointer arithmetic, and explain why.
- (ii) Explain how in some respects pointers are equivalent to arrays, and give one respect in which they differ.
- (alph\*) Explain why a function might be declared virtual in a C++ superclass.
- (alph\*) (i) How does the use of `void *` pointer in C allow a form of polymorphism? Give an example function declaration using the `void *` pointer.



- (ii) What is the main problem with the use of void \*, and how does C++ improve on this? Give the improved function declaration in C++ for your example function in part (d)(i).
- (alph\*) (i) Why might it be useful to define a copy constructor for a C++ class? Give an example of a copy constructor for a simple class.
- (ii) Why might it be useful to explicitly define the assignment operator (=) for a C++ class? Give an example definition of the assignment operator for a simple class.

