

UNIVERSITY OF LONDON

GOLDSMITHS COLLEGE

Department of Computing

B. Sc. Examination 2016

IS52028B

Principles and Application of Programming (15 credit version)

Duration: 1 hour 30 minutes

Date and time:

---

*This paper is in two parts: part A and part B. You should answer all of part A and one question from part B. Part A carries 40 marks, and part B carries 30 marks. The marks for each part of a question are indicated at the end of the part in [.] brackets.*

*There are 70 marks available on this paper.*

**THIS PAPER MUST NOT BE REMOVED  
FROM THE EXAMINATION ROOM**

**Part A**  
Multiple choice

**Question 1** C++: Each question has one correct answer

(a) C++ is...

[4]

- i. compiled
- ii. interpreted
- iii. just in time compiled
- iv. None of the mentioned

(b) How is the parameter *a* in the following function passed?

```
void f(int & a){  
  
}
```

- i. by value
- ii. by reference
- iii. by const reference
- iv. by pointer

[4]

(c) Which of the following correctly iterates through all elements of the `std::vector<std::string>` `v`?

- i. 

```
for(auto i : v){  
    std::cout << i << std::endl;  
}
```
- ii. 

```
for(int i : v){  
    std::cout << i << std::endl;  
}
```
- iii. 

```
for(int i = 0; i < v.length; i++){  
    std::cout << v[i] << std::endl;  
}
```
- iv. 

```
for(auto i = 0; i < v.length; i++){  
    std::cout << v[i] << std::endl;  
}
```

[4]

(d) What is the type of variable a?

```
vector <int> v;  
auto a = v.begin();
```

- i. auto
- ii. int
- iii. an iterator type
- iv. a `unique_ptr` type

[4]

(e) What is `std::cin` used for?

- i. allocating memory
- ii. safely managing pointers
- iii. reading user input
- iv. writing data to files

[4]

(f) What is the error in this code?

```
int a, b;  
std::cin >> a >> b;
```

- i. b should be declared a `std::string`
- ii. you cannot have two `>>` operators on a single line
- iii. it should be `<<` instead of `>>`
- iv. nothing, the code is correct

[4]

(g) What does the following code do?

```
vector <int> v;  
v.push_back(1);  
v.push_back(2);  
v.push_back(3);  
for (int i = 1; i < 4; i++){  
    std::cout << v[i] << std::endl;  
}
```

- i. Prints the numbers 0-2
- ii. Prints the numbers 1-3
- iii. Throws an error
- iv. Don't know

[4]

(h) Which constructor is called when this variable (of class MyClass) is created?

```
MyClass c;
```

- i. no constructor
- ii. a constructor taking an int parameter
- iii. a default constructor
- iv. a copy constructor

[4]

(i) What do we declare with `float* p, p1;`

- i. A float pointer p and a float pointer p1.
- ii. A float pointer p and a float p1.
- iii. A float p and a float pointer p1.
- iv. A float p and a float p1.

[4]

(j) An implementation of an assignment operator should:

- i. Check for self-assignment
- ii. Perform memory-management
- iii. Check for self-assignment and Perform memory-management
- iv. None of the above

[4]

## Part B

**Question 2**     Pointers

(a) Give two advantages and two disadvantages of declaring a variable as each of the following [12]

- i. value
- ii. pointer
- iii. reference

(b) Explain the difference between stack and heap memory allocation [8]

(c) Look at the following code and describe how and when memory is allocated and deallocated for each of the variables and the heap memory associated with each variable. In your answer refer to heap and stack memory. [10]

```
#include <iostream>

int *makePointer(int size)
{
    int *p = new int[size];
    for (int i = 0; i < size; i++){
        p[i] = i;
    }
    return p;
}

int main(int argc, const char * argv[]) {
    int s = 5;
    int * a;
    a = makePointer(s);

    std::cout << a[2];
    delete a;

    return 0;
}
```

**Question 3**    vector

- (a) Describe the `std::vector` and how it is used. [5]
- (b) Give three advantages of using a `std::vector<int>` rather than an array (`int []`) [3]
- (c) What output does the following program produce? You can assume that the program compiles and executes without producing any errors.

```
void function(std::vector<std::string> s) {
    std::vector<std::string> words;
    auto iter = s.begin();
    while (iter != s.end()){
        words.push_back(*iter);
        iter++;
    }
    std::string result = "";
    while (!words.empty()) {
        result = result + " " + words.back();
        words.erase(words.end()-1);
    }
    std::cout << result << std::endl;
}

int main(int argc, const char * argv[]) {
    std::vector<std::string> s;
    s.push_back("here");
    s.push_back("comes");
    s.push_back("summer");
    function(s);
}
```

[8]



- (d) Write a complete program that reads in the words in the file "test.txt" and counts the number of their occurrences. It should then print each word and the number of its occurrences.

[14]

If the 'test.txt' contains the following:

```
studying studying at goldsmiths.  
working and studying at goldsmiths.
```

The output of the program should be:

```
studying 3  
at 2  
goldsmiths. 2  
working 1  
and 1
```