UNIVERSITY OF BRISTOL

December 2024

School of Computer Science

MINI MOCK theory examination for the Degree of Master of Science in Computer Science (conversion)

COMSM1201
Programming in C

TIME ALLOWED:
45 minutes

This paper contains **twenty-five** questions.
All questions will be marked.
All questions must be answered on the multiple choice answer sheet.
The maximum for this paper is **25 marks**.

Other Instructions

Calculators and notes are not permitted for this examination.

All questions in this examination will be computer marked, so it is crucial that you fill in the answer sheet carefully. Use a pencil (not pen) to fill in the answer sheet, and a pencil eraser to remove any mistakes you make.

Write your candidate name, title of the examination, unit code, date, desk number, and your student number on the answer sheet in the relevant boxes.

TURN OVER ONLY WHEN TOLD TO BEGIN WORK

1. Which header file must be included to use the pre-defined `strcpy()` function?

   a) stdio.h b) stdlib.h c) stdbool.h d) string.h e) assert.h

2. Which of the following functions is used to read input from the terminal?

   a) `fgetc()` b) `scanf()` c) `input()` d) `read()` e) `fgets()`

3. Which of the following lines of codes, attempting to include the user-written header file mydefs.h, will result in no compilation errors or warnings?

   a) `#include <mydefs.h>;`
   b) `#include "mydefs.h";`
   c) `#include 'mydefs.h';`
   d) `#include <mydefs.h>`
   e) `#include "mydefs.h"`

4. Which C data type is used to store enumerators?

   a) bool b) char c) char* d) float e) int

5. Which of the following is not a standard data type in C?

   a) long int b) float c) long float d) double e) long double

6. Which of the following is not a reserved word in the C programming language?

   a) const b) float c) push d) unsigned e) while

7. Which of the following statements about identifiers is true?

   a) An identifier can start with a digit.
   b) Special characters like # cannot be used in identifiers.
   c) Identifiers are case-insensitive.
   d) Keywords can be used as identifiers if enclosed in double quotes.
   e) Identifiers can contain spaces.

8. For the following code snippet, what will be printed to the terminal?

```
int a[2][2] = {{1, 4}, {2, 5}};
printf("%d\n", a[1][0]);
```

   a) `0` b) `1` c) `2` d) `4` e) `5`

9. What will the following code snippet result in?

```
int x = 10;
int y = 4;
float z = x/y;
printf("%1.1f\n", z);
```

   a) `1.1` will be printed to the terminal.
   b) `2.0` will be printed to the terminal.
   c) `2.5` will be printed to the terminal.

d) `z` will be printed to the terminal.

e) The code won't compile successfully due to a syntax error.

10. For the following code snippet, what will be printed to the terminal?

```
int a = 2;
int b = 2;
int c = 1;
printf("%d\n", !(a==b));
```

a) -1 b) 0 c) 1 d) 2 e) 3

11. For the following code snippet, what will be printed to the terminal?

```
printf("%d\n", 2&1)
```

a) 0 b) 1 c) 2 d) 4 e) 8

12. For the following code snippet, what will be printed to the terminal?

```
printf("%d\n", 1<<1)
```

a) 0 b) 1 c) 2 d) 4 e) 8

13. Which of the following code snippets will result in a compilation error?

a) `for (;;) { }`

b) `while (1) { }`

c) `do { } while (1);`

d) `repeat { } until (1);`

e) None of the above

14. What is the value of `j` after the following code snippet is execute?

```
int i = 2;
int j = 5;
switch (i){
   case 1:
      j--;
      break;
   case 2:
      j++;
   case 3:
   case 4:
      j++;
      break;
   case 5:
      j = 10;
   default:
      j++;
}
```

a) 5 b) 6 c) 7 d) 10 e) 11

15. The following code snippet is intended to loop through values of `i` and print `Hello, World!` to the terminal once, when `i` is equal to `x`. Which line of code would result in unintended behaviour?

```
int main(void){
   int x = 5;
   for(int i = 0; i < 10; i++){
      if(i = x){
         printf("Hello, World!\n");
      }
   }
}
```

a) `int main(void){`

b) `int x = 5;`

c) `for(int i = 0; i < 10; i++){`

d) `if(i = x){`

e) `printf("Hello, World!\n");`

16. Which of the following code snippets will result in different behaviour to the others?

a)
```
for(int num = 0; num < 5; num++){
   printf("Bristol\n");
}
```

b)
```
for(int num = 0; num <= 4; ++num){
   printf("Bristol\n");
}
```

c)
```
for(int num = 5; num <= 10; num++){
   printf("Bristol\n");
}
```

d)
```
for(int num = 10; num > 5; --num){
   printf("Bristol\n");
}
```

e)
```
for(int num = 5; num != 0; num--){
   printf("Bristol\n");
}
```

17. For the following code snippet, what will be printed to the terminal?

```
int func(int* a, int*b, int n){
   if(n >= 0){
      return a[n]*b[n]+func(a,b,n-1);
   }
   return 0;
}

int main(void){
   int a[] = {0,1,2,3};
   int b[] = {1,0,1,1};
   printf("%d\n", func(a, b, 3));
   return 0;
}
```

a) `3` b) `4` c) `5` d) `6` e) `7`

18. For the below code snippet, if the first print statement prints `0x1200` to the terminal, what will the second print statement print to the terminal?

```
int arr[10];
printf("%p\n", &arr);
printf("%p\n", &arr[5]);
```

a) `0x0` b) `0x5` c) `0x1200` d) `0x1204` e) `0x1205`

19. What does the following code snippet define `my_structure` to be?

```
struct my_structure{
    int counter;
    int* tokens[MAX_SIZE];
};
typedef struct my_structure my_structure;
my_structure* s[25];
```

a) A struct that contains an integer, and an array of `MAX_SIZE` pointers to integers.

b) A struct that contains an integer, and a pointer to an array of `MAX_SIZE` integers.

c) A struct that contains an integer, a pointer to an array of `MAX_SIZE` integers, and an array with 25 elements.

d) An array of 25 pointers to structs that contain an integer and an array of `MAX_SIZE` pointers to integers.

e) A pointer to an array of 25 structs that contain an integer, and a pointer to an array of `MAX_SIZE` integers.

20. Which of the following is not a cause of memory leaks in C?

a) Trying to access an index beyond the bounds of an array.

b) Overwriting a pointer before the memory it points to is freed.

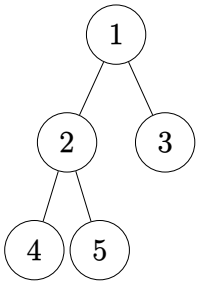
c) Opening a file and not closing it before the program ends.

d) Failing to free memory allocated using `calloc()`.

e) Failing to free memory allocated using `malloc()`.

21. What will the following snippet print when executed on the binary tree below?

```
void preorder(Node *root) {
    if (root == NULL) return;
    printf("%d ", root->data);
    preorder(root->left);
    preorder(root->right);
}
```



a) `1 2 4 5 3`
b) `4 2 5 1 3`
c) `1 3 2 5 4`
d) `1 4 5 2 3`
e) `4 5 2 3 1`

22. What is the worst case height of a binary tree with `n` nodes?

a) $O(n^2)$ b) $O(n \log n)$ c) $O(n)$ d) $O(\log n)$ e) $O(1)$

23. What is the worst case time complexity of a merge sort on an array?

a) $O(n^2)$ b) $O(1)$ c) $O(\log n)$ d) $O(n \log n)$ e) $O(n)$

24. Which of the following Abstract Data Types (ADTs) uses the Last In First Out (LIFO) approach?

a) Stacks b) Queues c) Sets d) Graphs e) Trees

25. Which of the following Abstract Data Types (ADTs) is best suited for the undo functionality in text editors?

a) Stacks b) Queues c) Sets d) Graphs e) Trees

THIS PAGE IS INTENTIONALLY LEFT BLANK

THIS PAGE IS INTENTIONALLY LEFT BLANK