UNIVERSIDADE DE AVEIRO

DEPARTAMENTO DE ELECTRÓNICA, TELECOMUNICAÇÕES E INFORMÁTICA

Programming Elements (2022/23)

Model Exam — November 2022 — Duration: 1h45m

NAME____ NUMBER_____________

I. (8 points) Insert your answers in the table below. For each question, only one of the answers is correct. Be aware that **each correct answer contributes with 0.8 points**, whereas **each incorrect answer deducts 0.2 points**.

Question	1	2	3	4	5	6	7	8	9	10
Answer										

I.1. The value of the expression `(a > 5 ? -1 : a++)` is:

A) `-1`, if `a` is equal to `5`

B) `-1`, if `a` is less than `5`

C) `-1`, if `a` is equal to `7`

D) `8`, if `a` is equal to `8`

E) This expression is invalid in C99

I.2. The assignment `a = a * 5 - 3` is equivalent to:

A) `a =* 5 - 3`

B) `a *= 5 - 3`

C) `a =* (5 - 3)`

D) `a *= (5 - 3)`

E) None of the previous

I.3. If the variable `x` has value `3`, then the expression `(x + x++)` has value:

A) `6`

B) `7`

C) `8`

D) Undefined in C99

E) The expression is invalid in C99

I.4. After the evaluation of expression `(x != 0 && (w = y / x) > 0)`, the variable `w` has a value:

A) Always equal to `(y / x)`
[✓ B)] Equal to `(y / x)`, if `x` is not zero
C) Always equal to the value it had before the evaluation of the expression
D) Undefined in C99
E) None of the previous answers

I.5. The expression `f(x++)`

A) Can always be used, independently of how `f` is implemented
B) Should never be used
C) Should not be used if `f` is a function
[✓ D)] Should not be used if `f` is a macro
E) None of the previous answers

I.6. The declaration `int (*a)[10]` means that the variable `a` is:

A) An *array* of 10 integers
B) An *array* of 10 pointers to integers
[✓ C)] A pointer to an *array* of 10 integers
D) A pointer to an *array* of 10 pointer t
E) This declaration is not valid in C99

I.7. The expression `('a' > 0)` is:

[✓ A)] Always true
B) Always false
C) True, if the variable `'a'` is greater than zero
D) True, if the variable `'a'` is `unsigned`
E) This expression is invalid in C99

I.8. The value of expression `(2 / 4 + 0.5)` is:

A) `1`
B) `1.0`
[✓ C)] `0.5`
D) `0`
E) Undefined in C99

I.9. The value of expression `(0x1 + 5)` is:

A) `1`
[✓ B)] `6`
C) `5`
D) `0`
E) The expression is invalid in C99

I.10. The C preprocessor directive `#if` has to be always terminated with:

A) `#else`
B) `#elif`
C) `}`
[✓ D)] `#endif`
E) None of the previous

II. Consider the following program:

```c
#include <stdio.h>

FILE *OpenFile(const char *name, const char *mode)
{
   FILE *fp = fopen(name, mode);

   if(!fp)
      fprintf(stderr, "Error while opening file \"%s\"\n", name);

   return fp;
}

void Copy(FILE *fpIn, FILE *fpOut)
{
   int c;

   while((c = getc(fpIn)) != EOF)
      putc(c, fpOut);
}

int main(int argc, char *argv[])
{
   FILE *fpIn, *fpOut;

   switch(argc)
   {
      case 2:
         if(!(fpIn = OpenFile(argv[1], "rb"))) return 1;
         Copy(fpIn, stdout);
         fclose(fpIn);
         break;

      case 3:
         if(!(fpIn = OpenFile(argv[1], "rb"))) return 1;
         if(!(fpOut = OpenFile(argv[2], "wb"))) return 1;
         Copy(fpIn, fpOut);
         fclose(fpIn);
         fclose(fpOut);
         break;
   }

   return 0;
}
```

1. (2 points) Briefly describe how the program works. **Note:** it is not intended a line-by-line description of the program's internal functioning, but a program's user manual, which should include, for example, what are the input data and how are they provided, what is the information provided by the program, what is the treatment given to error situations, etc.

2. (2 points) Suppose the program is named `copy`, and there is a file named `xpto`, which can be read but not written to. What is the result of the program when it is executed in the following ways:

a) `copy xpto`

b) `copy xpto xyz`

c) `copy ww xpto`

d) `copy xpto aaa bbb`

3. (2 points) Make a critical analysis of the presented program, pointing out deficiencies that you consider important and, at the same time, proposing solutions to overcome them.

III. (3 points) Without using the string manipulation functions available in the C library, write a function that duplicates a string. The prototype of this function should be

```
char *DupString(const char *str);
```

where `str` is the string to duplicate. The function should return a pointer to the new string, or `NULL` if it fails to do so.

IV. (3 points) Write a function that converts a non-negative integer into a string of zeros and ones with its binary representation. This function will have the prototype

```
char *DecToBin(unsigned int num);
```

where `num` is the number to convert. The function should return a pointer to the new string, or `NULL` if it fails to do so. The length of the string must be equal to the number of bits in an `unsigned int`, i.e., leading zeros must be represented.