University Center Abdelhafid Boussouf -Mila-
Institute of Science and Technology

Computer Science 2

2023/2024

الإمتحـان النهـائي

التمرين الأول (8.5ن):

1. ما هو الفرق بين الدالتين void func() و int func() ؟
2. ماذا يحدث عند محاولة الوصول إلى متغير من خارج نطاقة (outside its scope) ؟
3. ماذا يحدث في الذاكرة عندما نقوم بهذا الإعلان ; float a[5] ؟
4. ماذا نقصد ب Pass by reference ؟
5. باستعمال المؤشرات pointers أكتب لنا حلقة for التي تطبع لنا عناصر هذه المصفوفة ; int b[3]={1, 2,3} .
6. إذا قمنا بهذا الإعلان ; char str[]=" salam" ماذا سيعطينا هذا البيان ; printf("%c%c", str[0], str[4])؟ وكيف يسمى العنصر الأخير في هذه السلسلة (string)؟
7. لدينا المصفوفة التالية ; long Nin[5][4] ماهو حجم هذه المصفوفة وكم عدد البايتات(bytes) التي تشغلها في الذاكرة؟
8. ما هو الفرق بين عناصر المصفوفة array وعناصر البنية structure.

التمرين الثاني (5.5ن):

أكتب برنامجا في C لإنجاز ما يلي:

1- الإعلان عن مصفوفة ثنائية البعد $M_{3\times2}$ مع إرفاق القيم لعناصرها، بحيث $M = \begin{pmatrix} 1 & 2 \\ 5 & 6 \\ 8 & 7 \end{pmatrix}$

2- قم بطباعة العناصر التي قيمها أعداد زوجية.
3- قم بطباعة العناصر التي رقم سطرها فردي.

التمرين الثالث (6.5ن):

ماهي مخرجات هذا البرنامج:

```
1   #include <stdio.h>

2     void product(int a)
3     {
4        int i,b;
5        printf("--------------------------\n");
6        for(i=1; i<11; i++)
7        {
8           b=a*i;
9           printf("%dx%d=%d\n",a,i,b);
10       }
11        printf("--------------------------\n");
12    }
```

```
13  void allprod(int k, int m)
14    {
15       int j;
16       for(j=k; j<=m; j++)
17       {
18          product(j);
19       }
20    }
21  int main()
22  {
23     allprod(1,5);
24     return 0;
25   }
```

عمل موفق

University Center Abdelhafid Boussouf -Mila-
Institute of Science and Technology

Computer Science 2

2023/2024

تصحيح الإمتحان النهائي

التمرين الأول (8.5ن)

1- الدالة void func() ليست لها قيمة إرجاعية أي أننا لانكتب في تعريفها return ، كما يمكن استدعاؤها على شكل بيان داخل الدالة main()، وأما int func() فهي ترجع قيمة لذا نكتب داخل تعريفها return كما يتم استدعاؤها كقيمة لمتغير ما من الدالة main(). **(1.25ن)**

2- حدوث خطأ في البرمجة لأنه لم يتم التعرف على المتغير المعلن لأنه خارج نطاقه لتظهر لنا هذه العبارة " « ERROR « Undeclared Variable". **(0.75)**

3- عندما نقوم بهذا الإعلان ; float a[5] سيتم حجز خمسة أماكن في الذاكرة لخمس متغيرات مرتبة حسب الأندكس الخاص بها a[0], a[1], a[2], a[3], a[4]. سعة البيانات للمتغير الواحد هي 4 بايت ، ولكل متغير عنوانه الخاص في الذاكرة. **(0.5+0.5)**

4- "pass by reference" يعني النداء على الدالة باستعمال عنوان المتغير. ببساطة هو تمرير عنوان متغير الدالة عند الاستدعاء actual parameter الى متغير الدالة المستدعاة formal parameter. **(1ن)**

5- أولا يجب القيام بتعريف المؤشر بالاعلان عليه كالتالي ; int *Ptr بحيث ; Ptr=&b[0] ثم الإعلان على متغير int i ; لحلقة for ثم نكتب كالتالي **(1.5)**

```
for(i=0; i<3 ;i++)
  {
    printf("%d\n", *(Ptr+i));
  }
```

6- البيان يعطينا sm كما يسمى العنصر الأخير في السلسلة ب null character ويرمز له ب \0. **(1ن)**

7- حجم المصفوفة هو 4×5=20 عنصرا. بما أن عدد البيانات في العنصر الواحد هو 8 بايت فإن عدد البيانات في المصفوفة كلها هو 20×8=160 بايت. **(1ن)**

8- يكمن الفرق في أن عناصر المصفوفة لها نفس طبيعة البيانات لكن البنية عناصرها مختلفة من حيث طبيعة بياناتها. **(1ن)**

University Center Abdelhafid Boussouf -Mila-
Institute of Science and Technology

Computer Science 2

2023/2024

التمرين الثاني(5.5ن)

Program	*Output*

#include <stdio.h>
 	
int main() {
int M[3][2]={1,2,5,6,8,7};..... **(0.75)**
int i,j; **(0.25**
printf("Even elements:\n");
for(i=0; i<3 ;i++) **(0.5)**
{
for(j=0; j<2 ;j++) **(0.5)**
if(M[i][j]%2==0) **(1.25)**
{
printf("%d\n", M[i][j]);
}
}
printf("\nElements of the odd rows:\n");
for(i=0; i<3 ;i++) **(0.5)**
{
for(j=0; j<2 ;j++) **(0.5)**
if(i%2!=0) **(1.25)**
{
printf("%d\t", M[i][j]);
}
}
 	
return 0;}
	Even elements:
	2
	6
	8
	
	Elements of the odd rows:
	6 5

University Center Abdelhafid Boussouf -Mila-
Institute of Science and Technology

Computer Science 2

2023/2024

التمرين الثالث(6.5ن)

مخرجات البرنامج تتمثل في دالة تعطينا جدول الضرب للأعداد من 1 الى 5 كما يأتي **(6.5ن)**

Program	Output

#include <stdio.h>
 	
void product(int a)
{
int i,b;
printf("----------\n");
for(i=1; i<11; i++)
{
b=a*i;
printf("%dx%d=%d\n",a,i,b);
}
printf("----------\n");
}
void allprod(int k, int m)
{
int j;
for(j=k; j<=m; j++)
{
product(j);
}
}
int main()
{
allprod(1,5);
return 0;
}
	---------- 	
	1x1=1
	1x2=2
	1x3=3
	1x4=4
	1x5=5
	1x6=6
	1x7=7
	1x8=8
	1x9=9
	1x10=10
	---------- 	
	---------- 	
	2x1=2
	2x2=4
	2x3=6
	2x4=8
	2x5=10
	2x6=12
	2x7=14
	2x8=16
	2x9=18
	2x10=20
	---------- 	
	---------- 	
	3x1=3
	3x2=6
	3x3=9
	3x4=12
	3x5=15
	3x6=18
	3x7=21
	3x8=24
	3x9=27
	3x10=30
	---------- 	
	---------- 	
	4x1=4
	4x2=8
	4x3=12

	4x4=16 4x5=20 4x6=24 4x7=28 4x8=32 4x9=36 4x10=40 --- --- 5x1=5 5x2=10 5x3=15 5x4=20 5x5=25 5x6=30 5x7=35 5x8=40 5x9=45 5x10=50 ---