EXAMINATION INSTRUCTIONS

* Do not turn this page until asked to do so.
* Exam time is **75** minutes.
* Put the answers on the same question sheet, do not use any additional papers, even for scratch.
* Write your name, ID, section no. in the indicated places.
* Read the exam instructions.
* Read the honesty policy.
* Sign the following statement.

**Academic Integrity Policy**

Cheating in Exams is a violation of the Academic Integrity policy of AUC. Whispering, talking, looking at someone else’s paper, or copying from any source is considered cheating. Any one who does any of these actions or her/his answers indicates that she/he did any of them, will receive a punishment ranging from zero in this exam to failing the course. If repeated, it may lead to dismissal from AUC.

I have read the honesty policy and exam instructions and I am presenting this exam as entirely my effort.

Signature: _______________

---

**DO NOT USE THIS SECTION**

<table>
<thead>
<tr>
<th>Question</th>
<th>Points</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>100</strong></td>
<td></td>
</tr>
</tbody>
</table>
Question 1 (15 points)
A Perfect integer number is a positive integer number greater than 1 and whose sum of its factors (including 1) is equal to the number itself (for example, 6 is the first perfect number because \(1 + 2 + 3\) (the factors of 6) = 6).

The following C++ program takes a positive integer number greater than 1 and less than 1000 and prompts the user whether the entered number is perfect or not perfect. There are some missings (represented by dots) in the given program. Complete these missings such that the program could be compiled and run correctly. The program does not accept any integer number outside the given range (greater than 1 and less than 1000).

The Program

```cpp
#include <iostream>
using namespace std;
void main ()
{
    int num, sum;

    ..................

    do
    {
        cout << "Enter a positive integer greater than 1 and less than 1000: " << endl;
        cin >> num;

    } while (..............................);

    for ( int c = 1 ; ..................; c++ )

        if ( .........................)
            sum += ........ ;

        if ( ......................... )

            cout << "The number " << setw(4) << num << "is Perfect" << endl;

        else
            cout << "The number " << setw(4) << num << "is Not Perfect" << endl;

}
Question 2 (15 points)
A graduating student of AUC is awarded an honorary degree according to his/her final GPA. The honorary degree is granted according to the following rules:

<table>
<thead>
<tr>
<th>Final GPA</th>
<th>Honorary Degree</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.80 &lt;= GPA &lt;= 4.00</td>
<td>Highest Honors</td>
</tr>
<tr>
<td>3.60 &lt;= GPA &lt; 3.80</td>
<td>High Honors</td>
</tr>
<tr>
<td>3.40 &lt;= GPA &lt; 3.60</td>
<td>Honors</td>
</tr>
<tr>
<td>2.00 &lt;= GPA &lt; 3.40</td>
<td>Pass with no Honors</td>
</tr>
</tbody>
</table>

Draw a flow chart and write a program in C++ to input the GPA of a student and print out his/her honorary degree. **Show the three phases of software development: the analysis, design, and implementation. Implement your solution in C++ using switch structure.**

Enforce validation on the input GPA such that it is not less than 2.00 and not greater than 4.00.

### The Analysis

- **The Flow Chart**
The Program Using switch structure
### Question 3 (25 points)
Show the output of each of the following program segments:

<table>
<thead>
<tr>
<th>Program - 1</th>
</tr>
</thead>
</table>
| int x = 2, y = 2;  
while (x < 4)  
{  
y *= x++;  
cout << "x = " << setw(3) << x << " y = " << setw(3) << y << endl;  
} |

<table>
<thead>
<tr>
<th>Program - 2</th>
</tr>
</thead>
</table>
| int F[6] = {1, 1, 2, 3};  
for (int k = 4; k < 6; k++)  
{  
F[k] = F[k-1] + F[k-2];  
cout << setw(3) << k << setw(3) << F[k] << endl;  
} |

<table>
<thead>
<tr>
<th>Program - 3</th>
</tr>
</thead>
</table>
| const int ten = 10;  
int d;  
int n = 23059;  
do  
{  
d = n % 10;  
cout << d;  
n /= ten;  
} while (n != 0); |

<table>
<thead>
<tr>
<th>Program - 4</th>
</tr>
</thead>
</table>
| const int n = 6;  
int A[n] = {4, 4, 6, 8, 8, 9};  
bool flag = true;  
int k = 0;  
while ( (k < n-1) && (flag) )  
{  
if (A[k] > A[k+1])  
{  
flag = false;  
cout << setw(2) << k << setw(2) << A[k] << endl;  
}  
else  
k++;  
}  
if (flag)  
cout << " The list is OK" << endl; |

<table>
<thead>
<tr>
<th>Program - 5</th>
</tr>
</thead>
</table>
| int x = 1, y = 2;  
while (x < 3)  
{  
y /= ++x;  
cout << "x = " << setw(3) << x << " y = " << setw(3) << y << endl;  
} |

<table>
<thead>
<tr>
<th>Program - 6</th>
</tr>
</thead>
</table>
| int x, y = 40;  
for (x = 5; x > y; x *= 2)  
cout << setw(3) << x << setw(3) << y << endl;  
cout << "The Final Value Is: " << setw(5) << x / 2 % y; |
Question 4 (15 points)

Write a program fragment that uses *nested loops* to produce the following output:

```
$#\ldots\ldots#
$#
#$
$#
$#\ldots\ldots#
$#
#$
$#
$#\ldots\ldots#
$#
#$
$#
$#\ldots\ldots#
$#
#$
$#
$#\ldots\ldots#
$#
#$
$#
$#\ldots\ldots#
$#
#$
$#
```

Question 5 (10 points)

1. Rewrite the Boolean expression eliminating the not operator.

\[ \neg ((x \leq y) \lor (s > t)) \]

2. What is the value of the following expressions:

\[ (x - 5 \neq 5) \land (x - 5 = 5) \]

\[ (x - 5 \neq 5) \lor (x - 5 = 5) \]

3. Write down the final value of the following expression:

\[ (10 \% 4 \times 3 - 8 \geq 18 + 30 / 4 - 20) \land (7 = 7) \]
Question 6 (20 points)

Write a C++ program that allows the student of AUC to compute his/her GPA. The student has to enter to the program the following:

- The number of courses the student has taken.
- For each course, the letter grade the student has received in the course (the program should not accept any value for the letter grade other than A, B, C, D, and F) and the course credit hours (the program should not accept any value for the course credit hours other than 1, 2, and 3).

The numerical value of each letter grade is determined according to the following table:

<table>
<thead>
<tr>
<th>Letter grade</th>
<th>Numerical value</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>4.0</td>
</tr>
<tr>
<td>B</td>
<td>3.0</td>
</tr>
<tr>
<td>C</td>
<td>2.0</td>
</tr>
<tr>
<td>D</td>
<td>1.0</td>
</tr>
<tr>
<td>F</td>
<td>0.0</td>
</tr>
</tbody>
</table>

The Program

...