DEEC
DEPARTAMENTO DE ENGENHARIA ELECTROTÉCNICA E DE COMPUTADORES

Programação de Computadores (2013/14)

Segunda folha de Problemas

Matéria abordada: Escrita de mensagens no ecrã. Caracteres e inteiros (cast implícito). Expressões numéricas. Atribuição. Tabelas.
Bibliografia: Capítulo 4 e 5 de [Oua03].

1. Em C++ os caracteres são representados pelo seu código numérico na tabela ASCII. Considere o programa que se segue:

```
1  //=================================================
2  // Caracteres e inteiros
3  // Autor: Nome Apelido              Data:2013/mm/dd
4  // Objectivo:  Calculando e mostrando caracteres
5  // Utilizacao: Execute que as mensagens aparecem
6  //=================================================
7
8  #include <iostream>
9  using namespace std;
10
11 // O tipo char ocupa 8 bits. Por omisão,
12 // se interpretado como um valor numérico, tem sinal (de -128 a +127)
13 // Mas podemos definir unsigned char
14 // e nesse caso o valor numérico é (de 0 a 255)
15
16 char letra1, letra2, letra3, letra4;
17 int  codigo;
18
19 int main()
20 {
21    letra1 = 'P';   // Letra P
22    letra2 =  81;   // Código decimal da letra Q (cast implícito)
23    letra3 =  0122; // Código octal da letra R (cast implícito)
24    letra4 =  0x53; // Código hexadecimal da letra S (cast implícito)
25
26    //  Mostra as letras:
27    cout << "\nMostra as letras: " << letra1 << letra2  << letra3
28         << letra4 << endl;
29
30    cout << "\nAs letras e os Códigos:";
31    codigo = letra2 ;   // Cast implícito de char para int
32    cout << "\nA letra \'"<< letra1 << "\' tem código ASCII " << int(letra1);
33    cout << "\nA letra \'"<< letra2 << "\' tem código ASCII " << codigo;
34    cout << "\nA letra \'"<< letra3 << "\' tem código ASCII " << letra3 + 0;
35    cout << "\nA letra \'"<< letra4 << "\' tem código ASCII " << 0x53;
36
37    // Passando para as letras seguintes do alfabeto
38    letra1 = letra3 + 1; // Cast implícito
39    letra2 = letra1 + 1; // Cast implícito
40    letra3 = letra2 + 1; // Cast implícito
41
```

```
42      cout << "\n\nNovamente as letras e os Códigos:";
43      codigo = letra2 ;   // Cast implícito de char para int
44      cout << "\nA letra \'"<< letra1 << "\' tem código ASCII " << int(letra1);
45      cout << "\nA letra \'"<< letra2 << "\' tem código ASCII " << codigo;
46      cout << "\nA letra \'"<< letra3 << "\' tem código ASCII " << letra3+0;
47      cout << endl;
48      return(0);
49  }
```

(a) Diga qual a saída que espera encontrar.
Responda a esta questão sem transcrever, compilar e executar o programa!

(b) Abra ficheiro f2e1.cpp na pasta CodigoFonte. Se não tiver esse ficheiro transcreva o programa anterior (se fizer Ctrl+C Ctrl+V provavelmente terá de reescrever as aspas simples) para um ficheiro de nome f2e1.cpp na pasta (directoria) `~/Desktop/aula2`. Compile, execute e verifique a sua resposta à alínea anterior.

(c) Nas linhas 32 e 44 do programa anterior o primeiro valor é apresentado como um carácter e o segundo como inteiro. Porquê?

(d) Nas linhas 33 e 45 do programa anterior o primeiro valor é apresentado como um carácter e o segundo como inteiro. Porquê?

(e) Nas linhas 34 e 46 do programa anterior o primeiro valor é apresentado como um carácter e o segundo como inteiro. Porquê?

2. Para converter uma letra minúscula na sua maiúscula não precisamos memorizar a tabela ASCII! Sabemos que a qualquer letra minúscula está a uma distância da letra `'a'` igual à distância que a sua maiúscula está da letra `'A'` (e vice-versa).

   Por isso, também podemos dizer que a distância entre qualquer letra minúscula e sua maiúscula é fixa.

```
//=================================================
// Caracteres e inteiros, maiúsculas
// Autor: Nome Apelido               Data:2013/mm/dd
// Objectivo:  Calculando e mostrando caracteres
// Utilizacao: Execute que as mensagens aparecem
//=================================================

#include <iostream>
using namespace std;

char letra = 'q';
char grande;

int main()
{
    //Obtém a maiúscula de letra
    grande = 'A'+ (letra - 'a'); // <=> grande = letra + ('A' - 'a');
    //  Mostra
    cout << "Mostra:\n";
    cout << letra << '\n' << grande << endl;
```

```
    return(0);
}
```

(a) Diga qual a saída que espera encontrar.
Responda a esta questão sem transcrever, compilar e executar o programa!

(b) Abra ficheiro f2e2.cpp na pasta CodigoFonte. Se não tiver esse ficheiro transcreva o programa anterior (se fizer Ctrl+C Ctrl+V provavelmente terá de reescrever as aspas simples) para um ficheiro de nome f2e2.cpp na pasta (directoria) `~/Desktop/aula2`. Compile, execute e verifique a sua resposta à alínea anterior.

3. (a) Escreva um programa (f2e3a.cpp), que:
   - Pede um carácter (A..Z) ao utilizador e armazena numa variável do tipo `char`. Para isso deverá escrever algo do tipo:
     `cout << "Introduza uma letra (A..Z):";  cin >> letra;`
     Assuma que o utilizador é "bem comportado" ou seja que introduz de facto um carácter no intervalo pedido.
   - Seguidamente envia para o ecrã informação acerca da letra introduzida na seguinte forma (assumindo que digitou um `F`):

     ```
     Digitou F o qual tem o código 70
     ```

   (b) Copie f2e3a.cpp para f2e3b.cpp.
   Modifique o programa, para que seja agora enviado para o ecrã a letra minúscula correspondente ao carácter digitado pelo utilizador.
   Sugestão: No problema 2 é feita a conversão de minúscula para maiúscula. Utilize uma aproximação do mesmo tipo para fazer a conversão de maiúsculas para minúsculas.

4. Faça um programa que permita calcular a área de um rectângulo. O programa deve pedir ao utilizador que indique o comprimento dos lados e de seguida deve mostrar a valor da área pretendida bem como os valor introduzidos.

   Não se esqueça de previamente construir um algoritmo que permita resolver o problema e de introduzir os comentários adequados.

5. Escreva um programa que permita visualizar os códigos numéricos (decimais) ASCII correspondentes a 5 caracteres consecutivos.

   A primeira dessas letras deve poder ser introduzida pelo utilizador do programa.

   Declare uma tabela/vector de caracteres, `char letras[5]`, guarde o carácter lido em `letras[0]`, e os seguintes em `letras[1],...,letras[4]`.

   Note bem: A tabela `letras` guarda caracteres mas não é uma *string* no estilo C porque não está devidamente terminada com um `'\0'`.

   Apresente o resultado da seguinte forma (caso o utilizador tenha digitado a):

```
letra | Dec.
    a | 97
    b | 98
    c | 99
```

```
d | 100
e | 101
```

Não se esqueça de imprimir 4 espaços antes de cada letra; note também que a separação entre a letra e o seu código é "␣|␣", em que "␣" representa um espaço.

6. (a) Declare uma string **no estilo C** capaz de armazenar uma linha de texto com no máximo 80 caracteres úteis.
   (b) Declare uma string **no estilo C++** capaz de armazenar uma linha de texto.
   (c) Escreva um programa que pede duas linhas de texto ao utilizador (com um máximo de 65 caracteres úteis) – utilize o *Formulário*. E seguidamente apresenta ao utilizador o texto introduzido na seguinte forma (assumindo que o texto entre entre `""` foi o introduzido pelo utilizador):

   ```
   Primeira linha: "Monte da Caparica"
   Segunda linha:  "Figueira da Foz"
   ```

7. Resolva novamente o problema 5 mas sem utilizar uma tabela.

8. Escreva um programa que permita calcular a média de 3 valores reais introduzidos pelo utilizador e que dê o resultado arredondado para o inteiro mais próximo.

   Guarde esses três reais numa tabela/vector, `double notas[3]`, e utilize os valores armazenados na tabela para calcular a média.

9. Escreva um programa que peça dois inteiros e indique o resto da divisão inteira do primeiro pelo segundo.

TPC Construa um programa que peça o nome próprio de uma pessoa indicando que este terá de possuir 5 caracteres (Exemplos: Carla, Joana, Paulo, Pedro), e que mostre no ecrã o nome totalmente em maiúsculas.

Assuma que o utilizador foi bem comportado, ou seja que de facto introduziu um nome próprio com exactamente 5 caracteres e que o primeiro carácter já é uma maiúscula.

(a) Faça uma implementação utilizando strings no estilo C.
(b) Faça uma implementação utilizando strings no estilo C++. Qual a dimensão mínima da tabela capaz de armazenar uma string com 5 caracteres úteis?

Exemplo: Se o nome introduzido tiver sido "Joana" o programa apresenta como mensagem final[1]:

```
O nome dado em maiusculas: JOANA.
```

Referências

[Oua03] S. Oualline. *Practical C++ Programming.* O'Reilly, 3rd edition, 2003.

[1] Este exercício, assim como os exercícios 5 e 7, são intencionalmente monótonos, para suscitar a necessidade de intruções de controlo que permitam repetir blocos de de código: os ciclos!