**نظام الآسكي (ASCII)**

**نظام الآسكي هو نظام ترميز عالمي تم إنشاؤه لتمثيل الرموز والحروف باستخدام قيمة عددية معينة. يتم تمثيل كل حرف أو رمز باستخدام 1 بايت (أي 8 بتات)، حيث يتم تخصيص رقم معين لكل حرف أو رمز.**

**كيف يعمل نظام الآسكي؟**

**عند كتابة الرمز مثلًا 'a' في الكود البرمجي، مثل C++، يتم تنفيذ الخطوات التالية:**

**المترجم (Compiler):**

**يقوم المترجم بتحويل الحرف 'a' إلى قيمة الآسكي الخاصة به.**

**في حالة الحرف 'a'، القيمة الآسكي في النظام العشري هي 97.**

**تحويل القيمة العشرية إلى ثنائية:**

**بعد حصول المعالج على القيمة العشرية 97، يقوم المعالج (CPU) بتحويل هذه القيمة إلى النظام الثنائي.**

**97 في النظام العشري تساوي 01100001 في النظام الثنائي**

**تخزين القيمة في الذاكرة:**

**يتم تخزين هذه القيمة 01100001 في الذاكرة (الذاكرة العشوائية)، حيث يمكن للمعالج الوصول إليها واستخدامها في العمليات الحسابية أو المنطقية.**

**الربط بين الحروف الكبيرة والصغيرة في الآسكي:**

**في نظام الآسكي، هناك فرق ثابت بين الحروف الكبيرة والصغيرة. هذا الفرق هو دائمًا 32.**

**لتحويل حرف كبير إلى حرف صغير:**

**نقوم بإضافة 32 إلى قيمة الحرف الكبير.**

**مثال: إذا كان لدينا الحرف 'A' (الذي له قيمة 65 في الآسكي) ونريد تحويله إلى الحرف الصغير 'a'، نضيف 32 إلى القيمة:65 + 32 = 97، والتي تمثل الحرف 'a'.**

**لتحويل حرف صغير إلى حرف كبير:**

**نقوم بطرح 32 من قيمة الحرف الصغير.**

**مثال: إذا كان لدينا الحرف 'a' (الذي له قيمة 97 في الآسكي) ونريد تحويله إلى الحرف الكبير 'A'، نخصم 32 من القيمة:97 - 32 = 65، والتي تمثل الحرف 'A'.**

#include<iostream>

#include<string>

using namespace std;

int main(){

// Q1 :: Small To Capetal

char ch2 = 'a' - 32;

cout<<ch2<<endl;

// Q2 :: Capetal To Small

char ch4 = 'A' + 32;

cout<<ch4<<endl;

// Q3 :: Delete the symbol from the text string and make the symbol space empty : "A/bd.ULLa=h"

// name.length () ==> returns the number of elements array

// Eximple : INPUT => "A/bd.ULLa=h" || OUTPUT ==>A bd Ulla h

string name = "A/bd.ULLa=h";

for (int i = 0; i < name.length () ; i++)

{

    if ((name[i] < 'A' || name[i] > 'Z'  ) && ( name[i] < 'a' || name[i] > 'z' ) )

    {

      name[i] = ' ';

    }

}

cout<<name;

}