

Quiz#0

姓名:_____ 學號:_____

992 資訊二 A, C++ 程式設計, 15-minute quiz.

Openbook: 你可以看所有你自己的參考資料，請不要使用電腦，請不要和同學談話或是交換資料。

1) 請定義 struct data 結構型態，其中包含兩個欄位，一個是 100 個字元的陣列 cdata，一個是整數 idata

Sol:

method 1: <pre>struct data { char cdata[100]; int idata; };</pre>	To define a struct variable x: <table border="0"> <tr> <td>for a C compiler</td> <td>for a C++ compiler</td> </tr> <tr> <td>struct data x;</td> <td>data x;</td> </tr> </table>	for a C compiler	for a C++ compiler	struct data x;	data x;
for a C compiler	for a C++ compiler				
struct data x;	data x;				
method 2: <pre>typedef struct datatag { char cdata[100]; int idata; } data;</pre>	<pre>data x;</pre>				
method 3: <pre>typedef struct { char cdata[100]; int idata; } data;</pre>	<pre>data x;</pre>				

2) 請寫一個 main() 函式，請運用 malloc() 函式配置一個 struct data 型態的指標變數 input

Sol:

```
method 1 (ANSI C or C++ compiler):
void main()
{
    struct data *input = (struct data *) malloc(sizeof(struct data));
}

method 1 (C++ compiler), method 2, method 3:
void main()
{
    data *input = (data *) malloc(sizeof(data));
}
```

3) 請寫一個 while 迴圈運用 scanf() 函式由鍵盤一個字元一個字元讀入十六進位字元並且存放到 input.cdata 字元陣列中直到出現非十六進位數字字元為止，鍵盤輸入資料範例如: 1F20b5

Sol:

```
Version 1:
char cbuf;
int i=0;
while (scanf("%c", &cbuf))
    if ( (cbuf>='0'&&cbuf<='9')||(cbuf>='a'&&cbuf<='f')||(cbuf>='A'&&cbuf<='F') )
        input->cdata[i++] = cbuf;
    else
```

```

        break;
input->cdata[i] = 0;

```

Version 2:

```

char cbuf;
int i=0;
scanf("%c", &cbuf);
while ( ( cbuf>='0'&&cbuf<='9')||(cbuf>='a'&&cbuf<='f')||(cbuf>='A'&&cbuf<='F') )
{
    input->cdata[i++] = cbuf;
    scanf("%c", &cbuf);
}
input->cdata[i] = 0;

```

Version 3:

```

char cbuf;
int i=0;
while (1) {
    scanf("%c", &cbuf);
    if ( (cbuf>='0'&&cbuf<='9')||(cbuf>='a'&&cbuf<='f')||(cbuf>='A'&&cbuf<='F') )
        input->cdata[i++] = cbuf;
    else
        break;
}
input->cdata[i] = 0;

```

- 4) 請寫一個 convert 函式，在 main() 函式中呼叫這個函式，傳入上述**指標**變數 input，轉換這個字元陣列內的資料成為整數並且存放在對應的 idata 欄位中，例如上述資料用十進位表示的整數為 2039989

Sol:

```

#include "string.h"
void convert(struct data *);
void main()
{
    ...
    convert(input);
    ...
}
int hexValue(char c)
{
    if (c>='0'&&c<='9')
        return c - '0';
    else if (c>='A'&&c<='F')
        return c - 'A' + 10;
    else if (c>='a'&&c<='f')
        return c - 'a' + 10;
    assert(0);
    return -1;
}
void convert(struct data *ptr)
{
    ptr->idata = 0;
    while (ptr->cdata[i])
        ptr->idata += ptr->idata * 16 + hexValue(ptr->cdata[i++]);
}

```

- 5) 請以 printf() 函式列印 input **指標**變數**所參考到的結構**中的兩個欄位

Sol:

```

printf("%s %d", input->cdata, input->idata); or printf("%s %d", (*input).cdata, (*input).idata);

```

- 6) 請釋放所配置的記憶體

Sol:

```
free(input);  
better practice  
if (input) free(input), input = 0;  
i.e.  
if (input!=0)  
{  
    free(input);  
    input = 0;  
}
```

Note 1: there is a compiler error for the following code

```
typedef struct  
{  
    char cdata[100];  
    int idata;  
} data;  
struct data x;
```

Note 2: the following typedef is not correctly used, but VC compiler does not complain about it

```
typedef struct data  
{  
    char cdata[100];  
    int idata;  
};
```

typedef in the above code does not define the new type *data*, instead, the compiler just ignore it

if you use `data x;`

to define a variable x in an ANSI C compiler, the compiler would complains that data is not a valid type.

Another example: `typedef int;` // the VC compiler also does not complain about this meaningless usage.

Note 3: it is very common to distinguish a **type name** from a **variable name** by Capitalization or understore

<pre>typedef struct { char cdata[100]; int idata; } Data; Data x;</pre>	<pre>typedef struct { char cdata[100]; int idata; } _data; _data x;</pre>	<pre>typedef struct { char cdata[100]; int idata; } data_t; data_t x;</pre>
---	---	---