

CSE 333 – Section 6

Quiz 1/Midterm Review

James Okada Johnny Yan

Department of Computer Science & Engineering
University of Washington

October 31, 2013

Header Guard

some.h

```
1 #ifndef _SOME_H_
2 #define _SOME_H_
3
4 // statements
5
6 #endif // _SOME_H_
```

error.h

```
1 #ifdef _ERROR_H_
2 #define _ERROR_H_
3
4 // statements
5
6 #endif // _ERROR_H_
```

Off-by-One

StrCopy.c

```
1 char *StrCopy(const char *src) {  
2     char *dest = (char*)malloc(strlen(src) + 1);  
3     strcpy(dest, src);  
4     return dest;  
5 }
```

StrCopyBuggy.c

```
1 char *StrCopyBuggy(const char *src) {  
2     char *dest = (char*)malloc(strlen(src));  
3     strcpy(dest, src);  
4     return dest;  
5 }
```

A discouraging error

NamedArray

```
1 typedef struct named_array_st {  
2     char *name;  
3     int *elements;  
4 } NamedArray;
```

CreateNamedArray

```
1 NamedArray *CreateNamedArray(char *name, int nElements) {  
2     // statements  
3     NamedArray *na = (NamedArray*)malloc(sizeof(NamedArray));  
4     na->name = name;  
5     // statements  
6 }
```

Array & Pointer Puzzle

Sample 1

```
1 #include <stdio.h>
2
3 void printCharPtr(char *p1) {
4     printf("sizeof(p1) = %d\n", (int) sizeof(p1));
5 }
6
7 void printIntPtr(int **p2) {
8     printf("sizeof(p2) = %d\n", (int) sizeof(p2));
9 }
10
11 int main(int argc, char **argv) {
12     char p1[10];
13     int *p2[10];
14
15     printf("sizeof(p1) = %d\n", (int) sizeof(p1));
16     printf("sizeof(p2) = %d\n", (int) sizeof(p2));
17     printCharPtr(p1);
18     printIntPtr(p2);
19
20     return 0;
21 }
```

Array & Pointer Puzzle

Sample 1 cont.

```
1 #include <stdio.h>
2
3 void printCharPtr(char p1[]) {
4     printf("sizeof(p1) = %d\n", (int) sizeof(p1));
5 }
6
7 void printIntPtr(int *p2[]) {
8     printf("sizeof(p2) = %d\n", (int) sizeof(p2));
9 }
10
11 int main(int argc, char **argv) {
12     char p1[10];
13     int *p2[10];
14
15     printf("sizeof(p1) = %d\n", (int) sizeof(p1));
16     printf("sizeof(p2) = %d\n", (int) sizeof(p2));
17     printCharPtr(p1);
18     printIntPtr(p2);
19
20     return 0;
21 }
```

Array & Pointer Puzzle

Sample 1 cont.

```
1 #include <stdio.h>
2
3 void printCharPtr(char p1[10]) {
4     printf("sizeof(p1) = %d\n", (int) sizeof(p1));
5 }
6
7 void printIntPtr(int *p2[10]) {
8     printf("sizeof(p2) = %d\n", (int) sizeof(p2));
9 }
10
11 int main(int argc, char **argv) {
12     char p1[10];
13     int *p2[10];
14
15     printf("sizeof(p1) = %d\n", (int) sizeof(p1));
16     printf("sizeof(p2) = %d\n", (int) sizeof(p2));
17     printCharPtr(p1);
18     printIntPtr(p2);
19
20     return 0;
21 }
```

Array & Pointer Puzzle cont.

Sample 2

```
1 #include <stdio.h>
2
3 int main(int argc, char **argv) {
4     int incompArray[];
5     int compArray[5];
6     int *ptrArray = compArray;
7
8     printf("address of (incompArray) = %p\n", incompArray);
9     printf("address of (&incompArray) = %p\n", &incompArray);
10    printf("address of (compArray) = %p\n", compArray);
11    printf("address of (&compArray) = %p\n", &compArray);
12    printf("address of (ptrArray) = %p\n", ptrArray);
13    printf("address of (&ptrArray) = %p\n", &ptrArray);
14
15    return 0;
16 }
```


Array & Pointer Puzzle cont.

Sample 2 cont.

```
1 #include <stdio.h>
2
3 int main(int argc, char **argv) {
4     int incompArray[] = {1,2};
5     int compArray[5];
6     int *ptrArray = compArray;
7
8     printf("address of (incompArray) = %p\n", incompArray);
9     printf("address of (&incompArray) = %p\n", &incompArray);
10    printf("address of (compArray) = %p\n", compArray);
11    printf("address of (&compArray) = %p\n", &compArray);
12    printf("address of (ptrArray) = %p\n", ptrArray);
13    printf("address of (&ptrArray) = %p\n", &ptrArray);
14
15    return 0;
16 }
```

Content

Covered

- C: all the stuff taught in this quarter
- C++: class definition with simple constructor/destructor, including copy constructor and assignment operator

NOT covered

- Inheritance, general operator overloading, friend, anything else we haven't touched yet

Q&A

Any questions? In the sample midterm or anything else.