#  String Homework

1. The following function calls supposedly write a single new-line character, but some are incorrect. Identify which calls don’t work and explain why.

1. printf("%c", '\n');

Yes.

1. printf("%c", "\n");

No. Cannot output a string with a char format specifier.

1. printf("%s", '\n');

No. Cannot output a char with a string format specifier.

d) printf("%s", "\n");

Yes.

e) printf('\n');

No. Missing char format specifier.

f) printf("\n");

Yes.

g) putchar('\n');

Yes.

h) putchar("\n");

Can only output a char not a string.

i) puts('\n');

Can only output a string not a char.

j) puts("\n");

Yes.

k) puts("");

Yes.

2. Suppose that p has been declared as follows:

char \*p = "abc";

Which of the following function calls are legal? Show the output produced by each legal call, and explain why the others are illegal.

a) putchar(p); Illegal. Cannot output string.

b) putchar(\*p); Legal. Output: a

c) puts(p); Legal. Output: abc

d) puts(\*p); Illegal. Cannot output char.

3. Suppose that str is an array of characters. Which of the following statements is not equivalent to the other three?

1. \*str = 0;

b) str[0] = '\0';

c) strcpy(str, "");

d) strcat(str, ""); Not equivalent.

4. What will be the value of the string s1 after the following statements have been executed?

strcpy(s1, "computer");

strcpy(s2, "science");

if (strcmp(s1, s2) < 0)

strcat(s1, s2);

else

strcat(s2, s1);

s1[strlen(s1)-6] = '\0';

computers

5. The following function supposedly creates an identical copy of a string. What’s wrong with the function?

char \*dup(char \*p) {

char \*q;

strcpy(q, p);

return q;

}

No storage is allocated for q.

6. What does the following program print?

#include <stdio.h>

int main(void) {

char s[] = "Hello";

char \*p;

for (p = s; \*p; p++)

--\*p;

puts(s);

return 0;

}

Gdkkn