Used as CS 202 f2017 examA. Make a directory ~/EXAM/examA/ in your cs202 account.
Create the programs as examA_1.c, examA_2.c, etc.

ISU Programming Assessment *Sample*

~~Name: _____~~ ~~CS class account: _____~~

~~Put all answers in the boxes. Nothing you write outside of the boxes will be counted. Did you bring an eraser?~~

1. Write a C program that prints prints `h` many lines, with the i^{th} line being `length[i]` many `'-'` characters.

```
int main(int argc, char *argv[]) {
```

```
  const int h = 30;
  int lengths[h];
  for(int i=0; i < h; i++) lengths[i] = sin(3.14*i/h)*h*2;
```

```
  return 0;
}
```

2. Write a C program that reads from stdin one character at a time and prints the characters in "excited mode" - all '.' characters are printed as '!', letters are printed as upper case, and all other characters are printed as is.

```
int main(int argc, char *argv[]) {
```

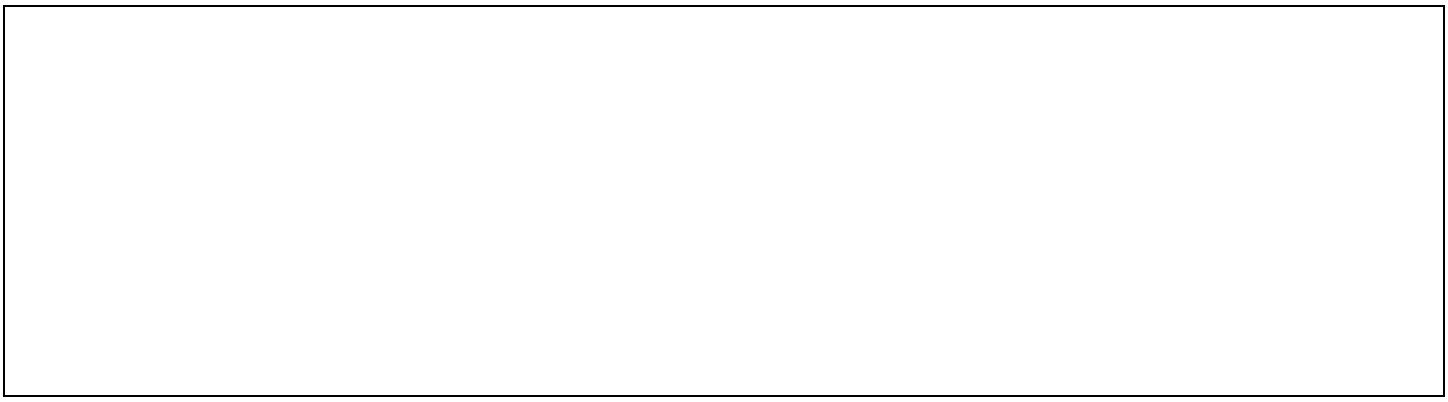
```
  return 0;
}
```

Files to start with and correct compiled programs are in ~jkinne/public_html/cs202-f2017/EXAM/examA/
Grading: each program full, ⅔, ½, ⅓, or 0 credit. Best program 12 pts, then 6, 6, 3, 3.

Used as CS 202 f2017 examA. Make a directory ~/EXAM/examA/ in your cs202 account.
Create the programs as examA_1.c, examA_2.c, etc.

3. Write a loop that finds the longest word in the linked list. Use the variables declared below.

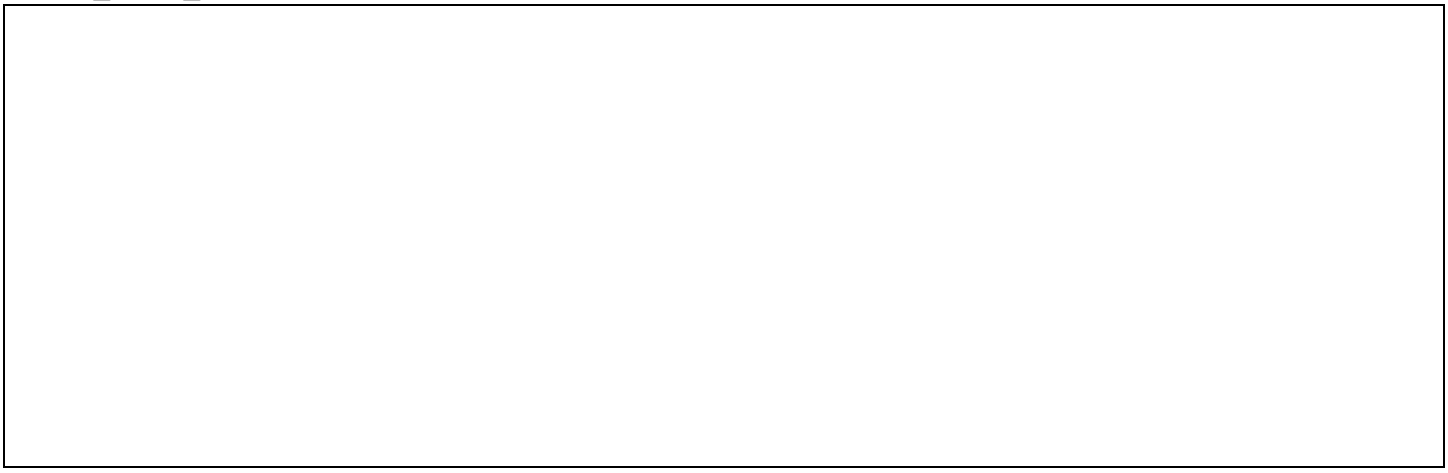
```
typedef struct WORD_T {
  char *w;   // space for the word
  int l;     // length of w, so we don't have to recompute
  int count; // for frequency counts
} word_t;

typedef struct LIST_NODE_T {
  word_t * data;
  struct LIST_NODE_T *prev, *next;
} list_node_t;

int main(int argc, char *argv[]) {
  list_node_t *head, *ptr;
  /* Assume that the list is somehow created here.  Use examA_3.c starter file
*/
```



```
  return 0;
}
```

4. Write a function name `height` that has the root of a binary tree as parameter, and which returns the height of the tree (maximum length from root to a leaf). Use the following declaration.

```
typedef struct BST_NODE_T {
  word_t * data;
  struct BST_NODE_T *left, *right;
} bst_node_t;
```



Files to start with and correct compiled programs are in ~jkinne/public_html/cs202-f2017/EXAM/examA/
Grading: each program full, ⅔, ½, ⅓, or 0 credit. Best program 12 pts, then 6, 6, 3, 3.

Used as CS 202 f2017 examA. Make a directory ~/EXAM/examA/ in your cs202 account.
Create the programs as examA_1.c, examA_2.c, etc.

5. Write a C function named `roundPower2` that has one long int as a parameter. The function should return the next power of 2 the next is <= the parameter. For example, if the parameter is 17 the function should return 16. If the parameter is 100 the function should return 64. If the parameter is 8 the function should return 8.