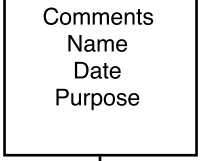
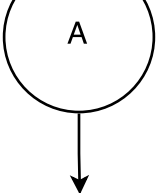

Header File
Essay.h

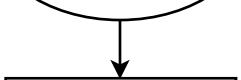
Essay()

void setSpell(

float spell)
float getGram() const
float getLength() const


Comments
Name
Date
Purpose

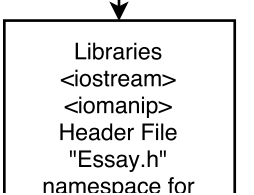

A

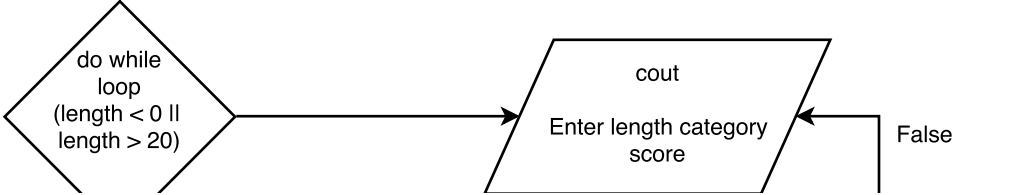

Declare class
Essay:public


grammer = 0.0


spelling =


float getLength() const
spell


Libraries
<iostream>
<iomanip>
Header File
"Essay.h"
namespace for

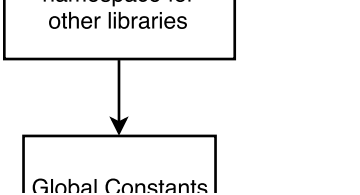

do while
loop
(length < 0 ||
length > 20)
cout
Enter length category
score
False

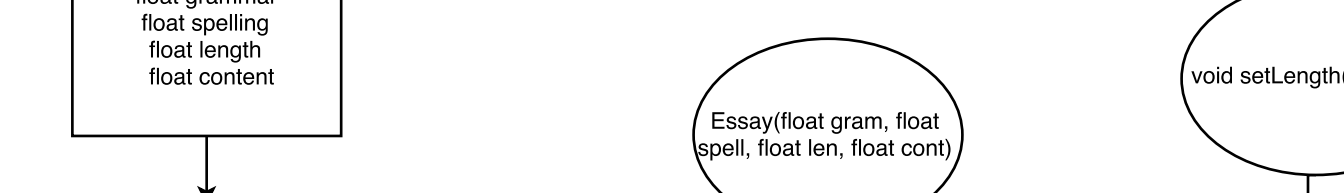
GradedActivity

Protected:
Variables
float grammar

spelling = 0.0
length = 0.0
content = 0.0


return
grammer
return
length

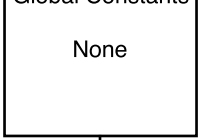

other libraries
Global Constants

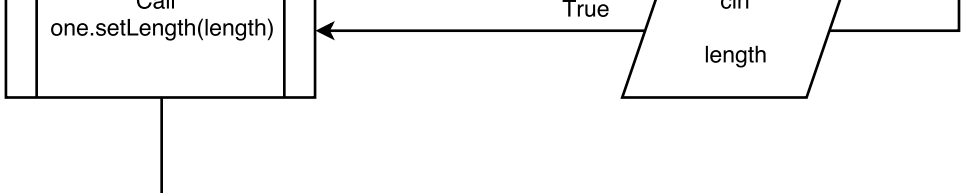

float grammar
float spelling
float length
float content
Essay(float gram, float spell, float len, float cont)
void setLength

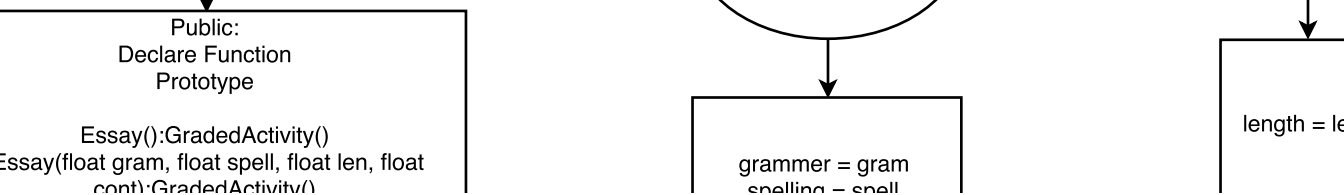
(float len)

float getSpell() const

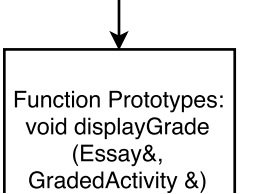
float getContent() const

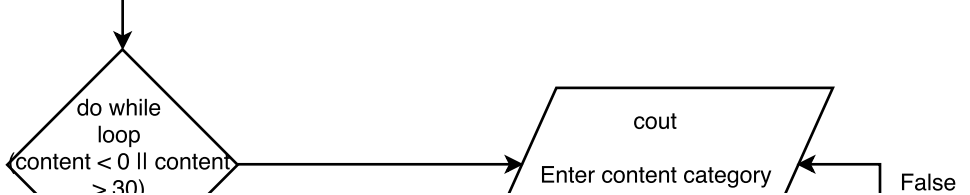

Global Constants
None

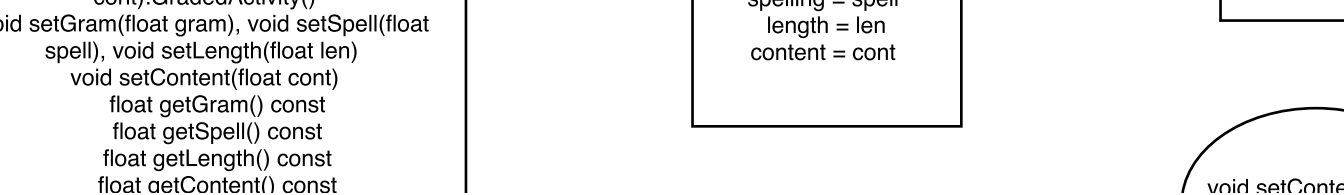

Call
one.setLength(length)
True
cin
length

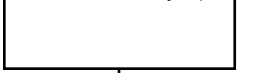

Public:
Declare Function
Prototype
Essay():GradedActivity()
Essay(float gram, float spell, float len, float
grammer = gram
length = le

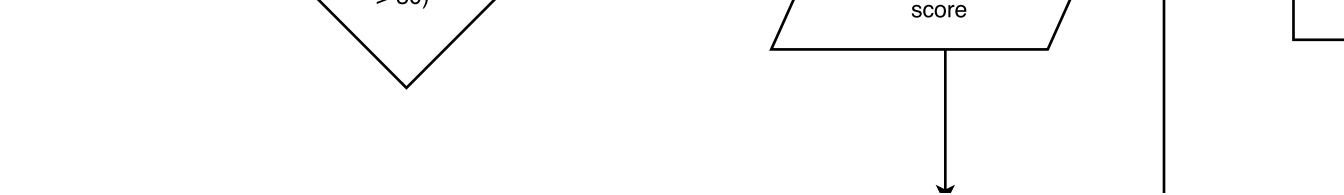

en
return
spelling
return
content


Function Prototypes:
void displayGrade
(Essay&,
GradedActivity &)


do while
loop
(content < 0 || content
> 30)
cout
Enter content category
False

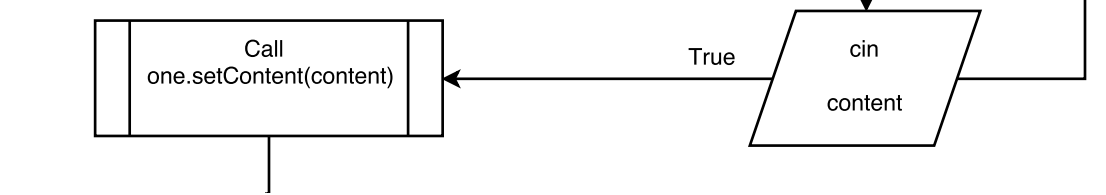

oid setGram(float gram), void setSpell(float
spell), void setLength(float len)
void setContent(float cont)
float getGram() const
float getSpell() const
float getLength() const
float getContent() const
spelling = spell
length = len
content = cont
void setConte

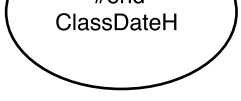


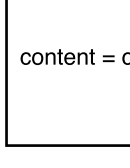

score


#end
void setCont(
cont)

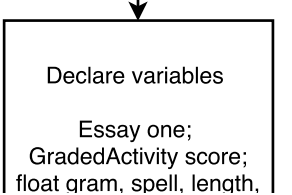

Main
Display Essay
Grade

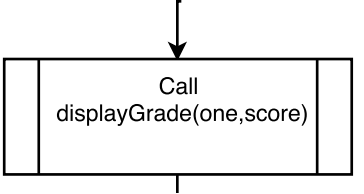

Call
one.setContent(content)
True
cin
content

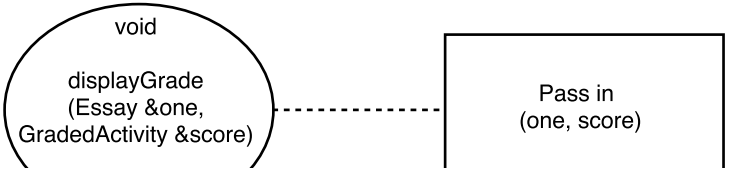

ClassDateH

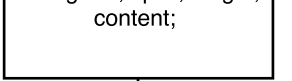

content = c

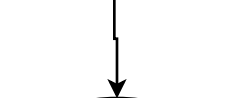
cont

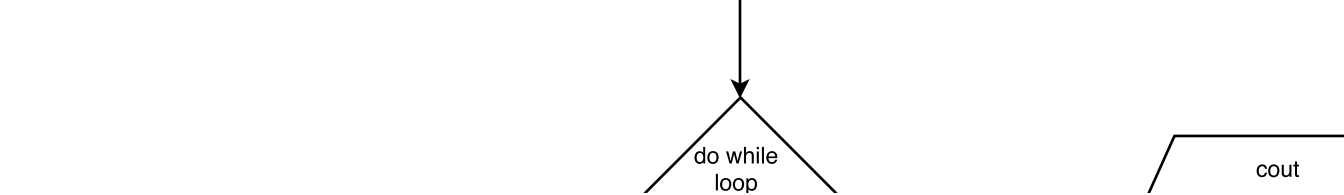

Declare variables
Essay one;
GradedActivity score;
float gram, spell, length,

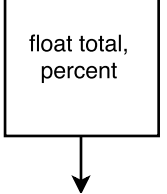

Call
displayGrade(one,score)


void
displayGrade
(Essay &one,
GradedActivity &score)
Pass in
(one, score)


content;


Exit Stage
Right

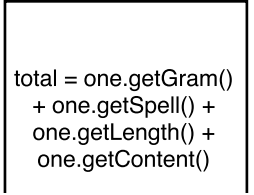

do while
loop
cout


float total,
percent

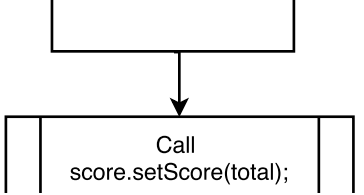

(gram < 0 ll
gram > 30)
Enter grammar cat
score

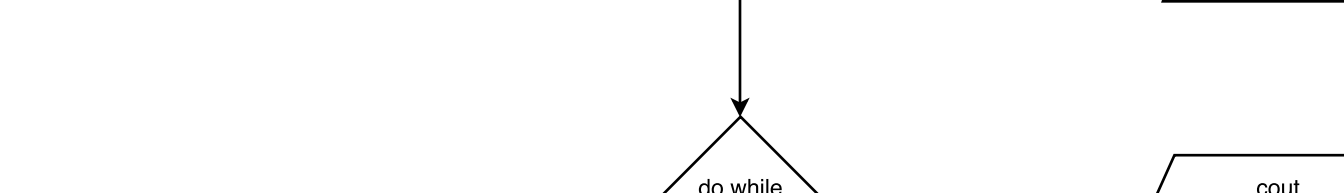
egory

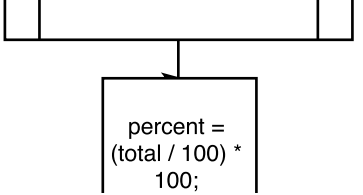
False

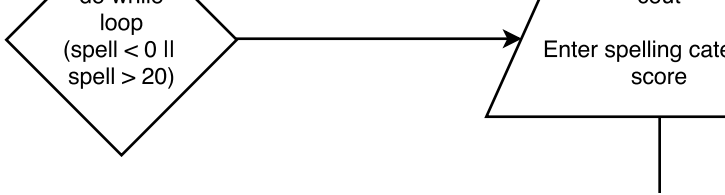

total = one.getGram()
+ one.getSpell() +
one.getLength() +
one.getContent()

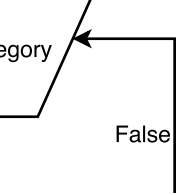

Call
one.setGram(gram)
True
cin
gram

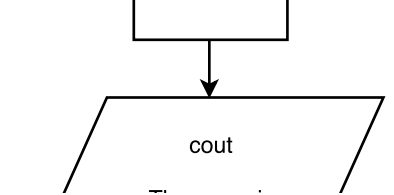

Call
score.setScore(total);


do while
cout

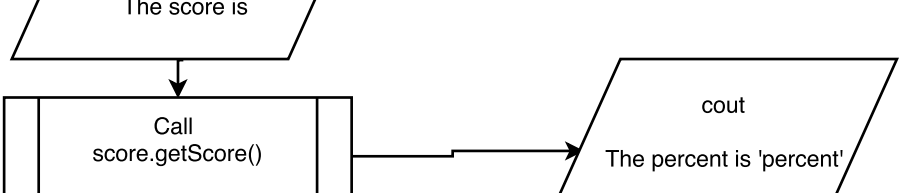

percent =
(total / 100) *
100;

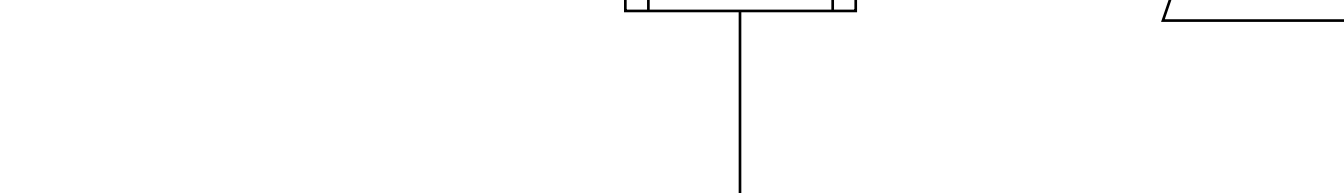

do while
loop
(spell < 0 ||
spell > 20)
Enter spelling cate
score


egory
False

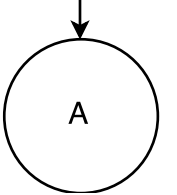

cout


Call
one.setSpell(spell)
True
cin
spell


The score is
Call
score.getScore()
cout
The percent is 'percent'



/



A