

Third Year --- Computers: Data Structures March 2011

Sheet 1: ADT's and C++ Classes

- 1) Write a class Time ADT that captures the time of day represented using a single integer representing minutes from the time 00:00 AM. The time ADT should support the operations: a) readTime that reads time in the format HH:MM AM/PM where HH is the hour and MM is the minutes b) subtractTimes(Time t) that provides the difference between the current time object and object t.
- 2) Write a program that prompts the user for a start time, a stop time, and a distance traveled (in miles) and computes the speed in miles per hour. Use the Time ADT.
- 3) Add a member function DisplayTime to the Time ADT that will display the stored time in format HH:MM:AM_PM.
- 4) Write a class person that contains the variable name, that is a character array, an integer representing the age, a float representing the salary. Then write a program which uses that class to read the data of 20 persons and print the name of each person and get the total salaries.
- 5) Write a class polynomial where the polynomial is a sum of terms having the form a x^{e} . The largest exponent is called its degree. Coefficients that are zero are not displayed. The class must contain the following functions:

void Zero()	polynomial p(x)=0
bool IsZero()	if(poly $\neq 0$) return false, else return true
float Coef(expon)	if(expon ε poly) return its coefficient, else return zero
int Lead_Exp()	returns the largest exponent in the polynomial
bool Attach(Coeff,expon)	if (expon ε poly) return error, else insert the term
((((((Do we add a function to display the polynomial?	
void Display() D	Displays the polynomial))))))

- 6) Write an ADT IntVector that can be used to store a vector of integers. Operations should include v.put(i,j), which puts the value j into position i in the vector, and v.at(i) which returns the value located at position i. The size of the vector is equal to 100. Your vector class should perform range checking.
- 7) Programming Assignment: Write an ADT for a fractionl number that has a numerator and denumerator (A fraction is the form X/Y where X and Y are integers). The class has the member functions add, subtract, multiply, divide, and simplify. The first four functions obviously perform the respective operation on two fractions one of which is this fraction and the other is passed as parameter to the function, the result of the operation is placed in a second parameter passed to the function. The member function simplify will put the fraction in the simplest form. For example if the fraction resulting from one of the above operations is 16/24, the function simplify should put the result as 2/3. Implement and test your fraction ADT on the compiler of your choice (the recommended compiler is DevC++).