Raphael Restrepo

Patela

Homework 4

1.

enum Planet {MERCURY, VENUS, EARTH, MARS, JUPITER, SATURN, URANUS, NEPTUNE, PLUTO};

2.

Planet whatPlanet(string planet) {

if(planet == "mercury") {

return MERCURY;

} else if(planet == "venus") {

return VENUS;

} else if(planet == "earth") {

return EARTH;

} else if(planet == "mars") {

return MARS;

} else if(planet == "jupiter") {

return JUPITER;

} else if(planet == "saturn") {

return SATURN;

} else if(planet == "uranus") {

return URANUS;

} else if(planet == "neptune") {

return NEPTUNE;

} else if(planet == "pluto") {

return PLUTO;

} else {

return EARTH;

}

}

3.

string whatPlanet2(Planet planet) {

if(planet == MERCURY) {

return "Mercury";

} else if(planet == VENUS) {

return "Venus";

} else if(planet == EARTH) {

return "Earth";

} else if(planet == MARS) {

return "Mars";

} else if(planet == JUPITER) {

return "Jupiter";

} else if(planet == SATURN) {

return "Saturn";

} else if(planet == URANUS) {

return "Uranus";

} else if(planet == NEPTUNE) {

return "Neptune";

} else if(planet == PLUTO) {

return "Pluto";

} else {

return "Error";

}

}

4.

for(int i = MERCURY; i != PLUTO; i++) {

cout << whatPlanet2((Planet)i) << endl;

}

5.

struct Time {

int minutes;

int seconds;

};

6.

Time someTime;

someTime.minutes = 6;

someTime.seconds = 54;

7.

string topTenList[10];

enum COLORS {RED, ORANGE, YELLOW, GREEN, BLUE, INDIGO, VIOLET};

float colorMix[VIOLET];

enum DAYS {SUNDAY, MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, SATURDAY};

struct activity {

int day;

string activity;

};

int month [6][7];

8.

struct DataSet {

float input[5];

float output[5];

float working[5];

};

9.

DataSet Set[3];

10.

for(int i = 0; i < 3; i++) {

for(int v = 0; v < 5; v++) {

Set[i].input[v] = 0.0;

Set[i].output[v] = 0.0;

Set[i].working[v] = 0.0;

}

}

11.

bool Equals(const DataSet arg1, const DataSet arg2)