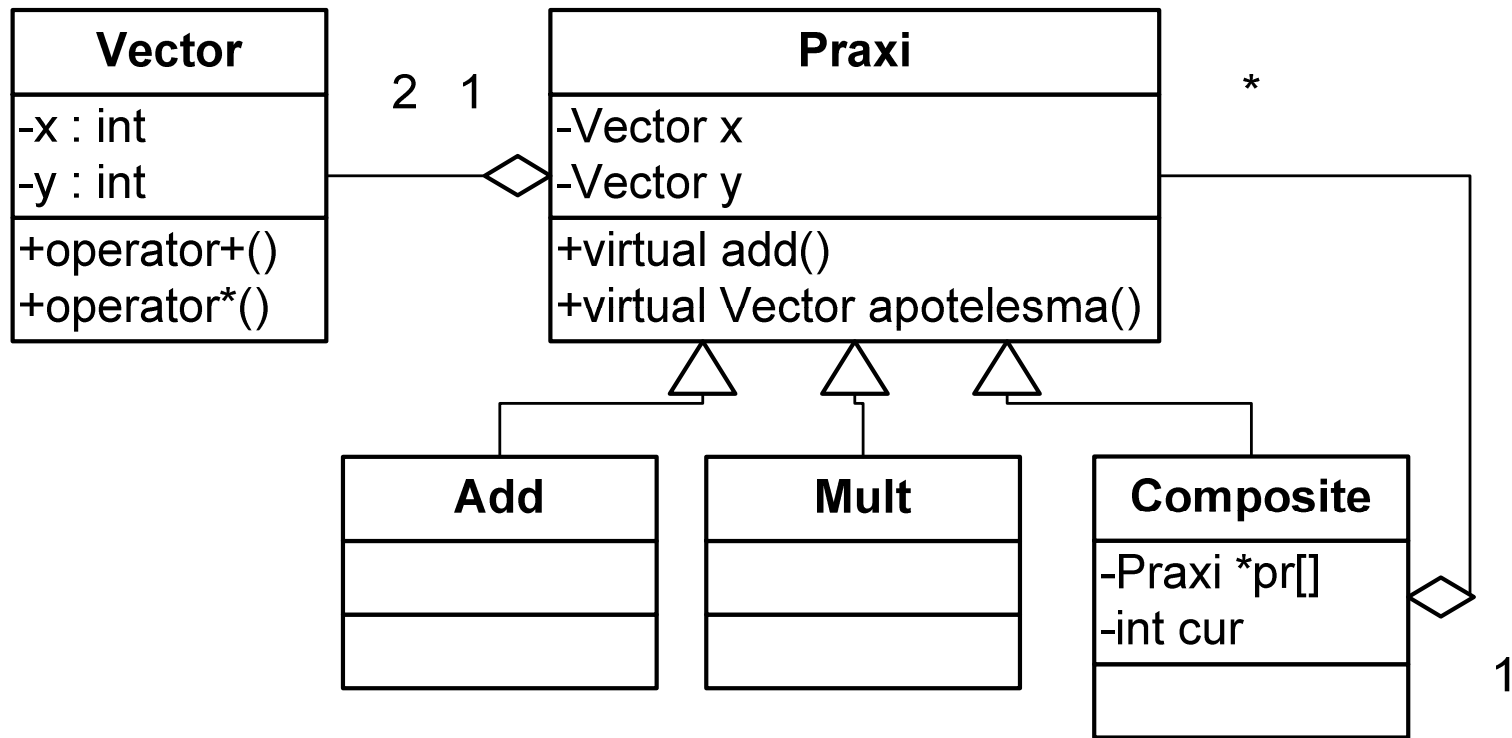


Πρότυπο Σύνθετο

- Αναπαράσταση και υπολογισμός αριθμητικών εκφράσεων.



Class Vector

```
class Vector {
    int x;
    int y;
public:
    Vector() {}
    Vector(int x_, int y_) {x=x_; y=y_;}
    Vector operator+(Vector w) {
        Vector temp;
        temp.x =x + w.x;
        temp.y =y + w.y;
        return temp;
    }
    Vector operator*(Vector w) {
        Vector temp;
        temp.x =x * w.x;
        temp.y =y * w.y;
        return temp;
    }
}

void print(){
    cout<<"x="<<x<<endl;
    cout<<"y="<<y<<endl;
}
```

Class Praxi

```
class Praxi {  
    protected:  
        Vector *v1;  
        Vector *v2;  
    public:  
        Praxi() {}  
        Praxi(Vector *v1_, Vector *v2_) {v1=v1_;v2=v2_;}  
        virtual void addPraxi(Praxi*) = 0;  
        virtual Vector apotelesma() = 0;  
};
```

Class Multi

```
class Multi : public Praxi
{
    public:
        Multi(Vector *v1, Vector *v2) : Praxi(v1,v2) {}
        void addPraxi(Praxi*) {}
        Vector apotelesma() {*v2 = (*v1)*(*v2); return *v2;}
};
```

class Add

```
class Add : public Praxi {  
    public:  
        Add(Vector *v1_, Vector *v2_) : Praxi(v1_,v2_) {}  
        void addPraxi(Praxi*) {}  
        Vector apotelesma() { *v2=(*v1)+(*v2); return *v2;}  
};
```

class Composite

```
class Composite : public Praxi {  
  
private:  
    Praxi* pr[10];  
    int cur;  
  
public:  
    Composite() : Praxi() {cur=0;}  
    void addPraxi(Praxi* p){  
        pr[cur]=p;  
        cur++;  
    }  
}
```


class Composite

```
Vector apotelesma() {  
    Vector temp;  
    int i;  
    for(i=0;i<cur;i++)  
        temp=pr[i]->apotelesma();  
    return temp;  
}  
};
```

Main

```
int main(){
    Vector temp;
    Praxi* comp = new Composite();
    Praxi* p;

    FILE *fp = fopen("input.txt","r");
    char c,telestis;
    int x,y;
    int cur=0;
    Vector **telestioi = new Vector* [1000];
    fscanf(fp,"%c%d,%d%c",&c,&x,&y,&c);
    telestioi[cur] = new Vector(x,y);
    cur++;
    fscanf(fp,"%c",&telestis);
    while(telestis!='\n'){
        fscanf(fp,"%c%d,%d%c",&c,&x,&y,&c);
        telestioi[cur] = new Vector(x,y);
        cur++;
    }
}
```

Main

```
if(telestis=='+')
    p = new Add(telesteoi[cur-2],telesteoi[cur-1]);
    if(telestis=='*')
        p = new Multi(telesteoi[cur-2],telesteoi[cur-1]);
    comp->addPraxi(p);
    fscanf(fp,"%c",&telestis);
}
temp = comp->apotelesma();
temp.print();

fclose(fp);
```

Input.txt

- $(2,3)+(6,10)*(2,3)$