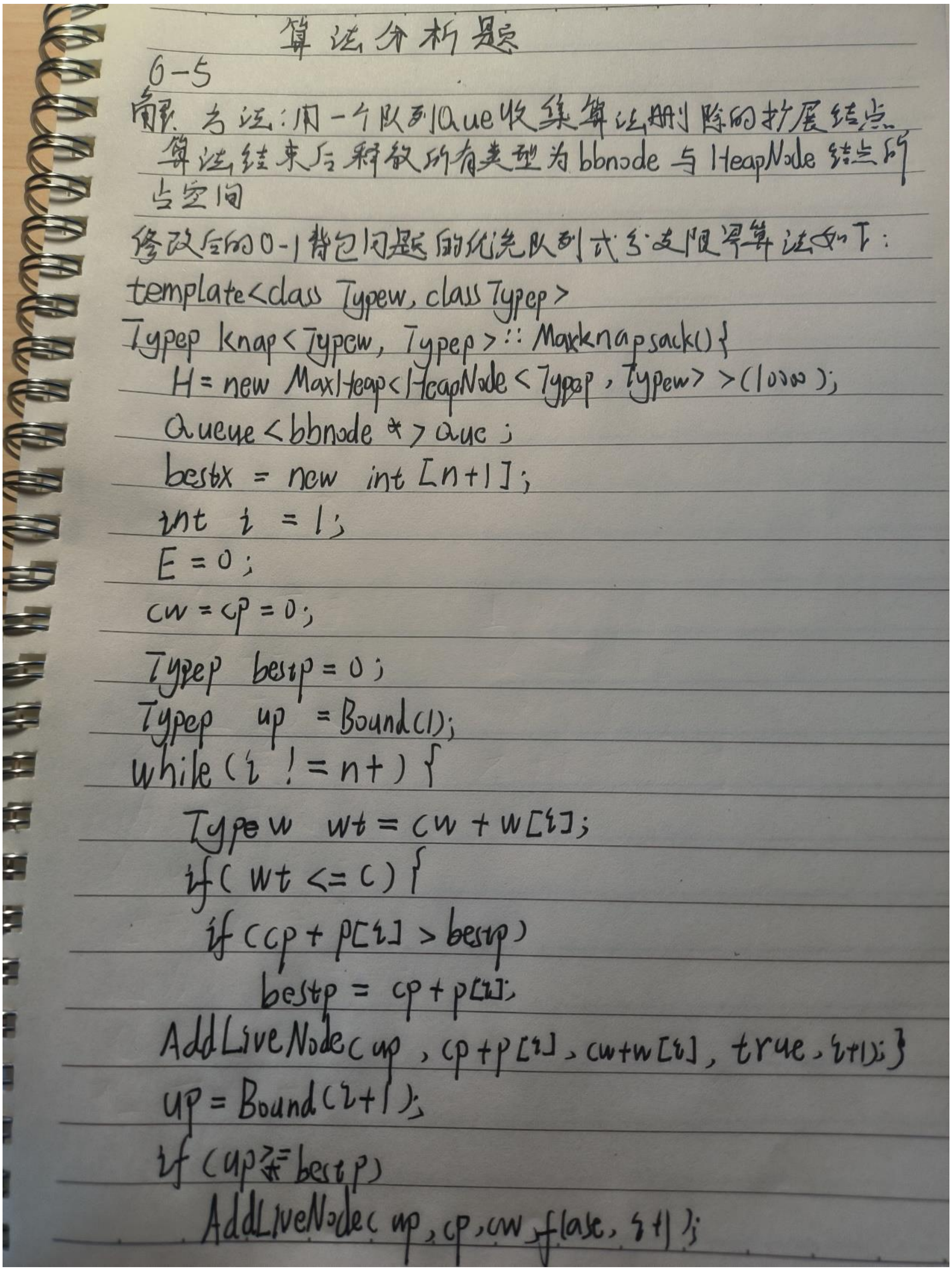
第六章算法分析题

学号：2209060322　　姓名：梁桐　　班级：计算机 2203



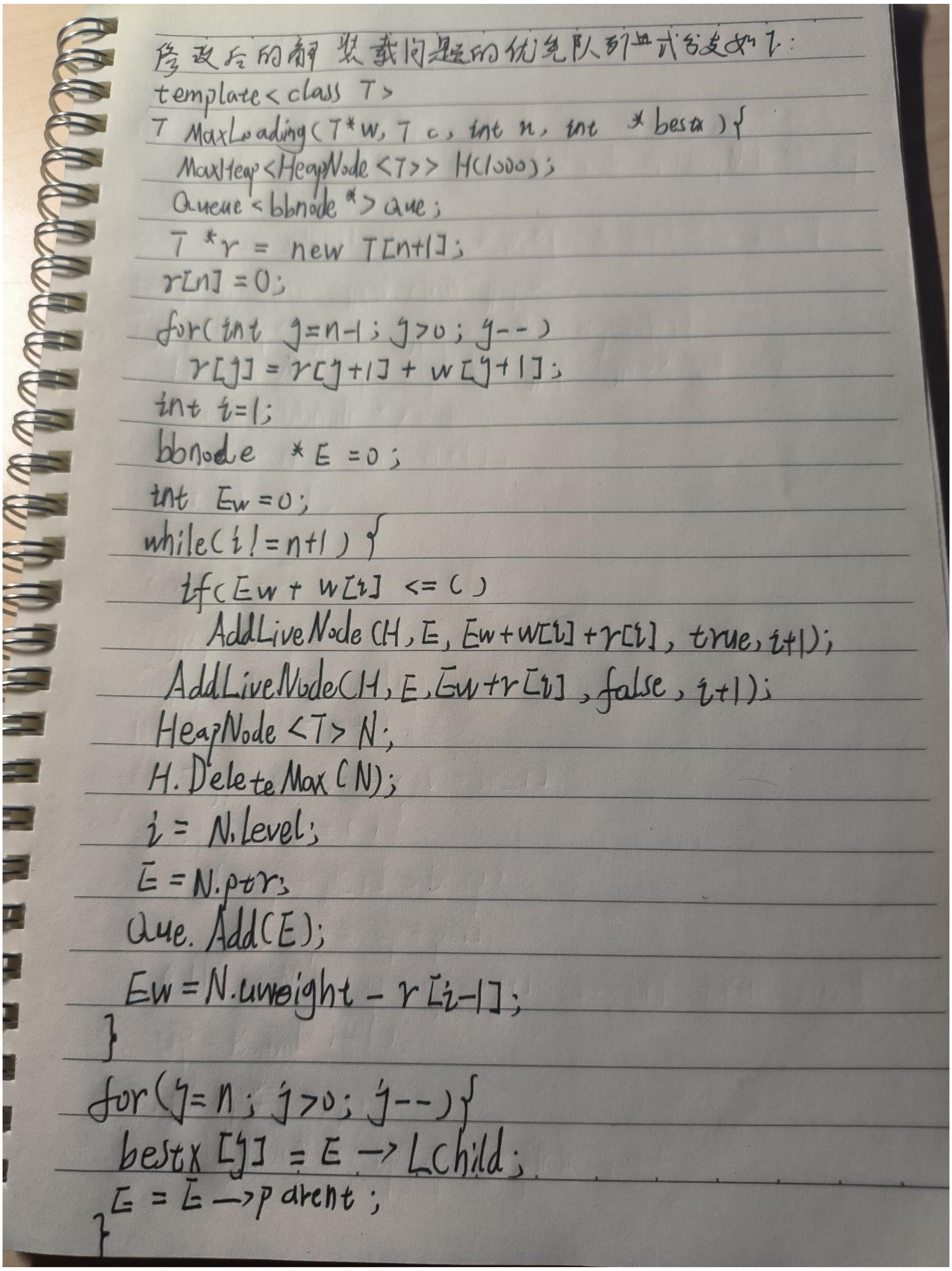
算法分析题

6-5

解：方法：用一个队列Que收集算法删除的扩展结点。算法结束后释放所有类型为bbnode与HeapNode结点所占空间

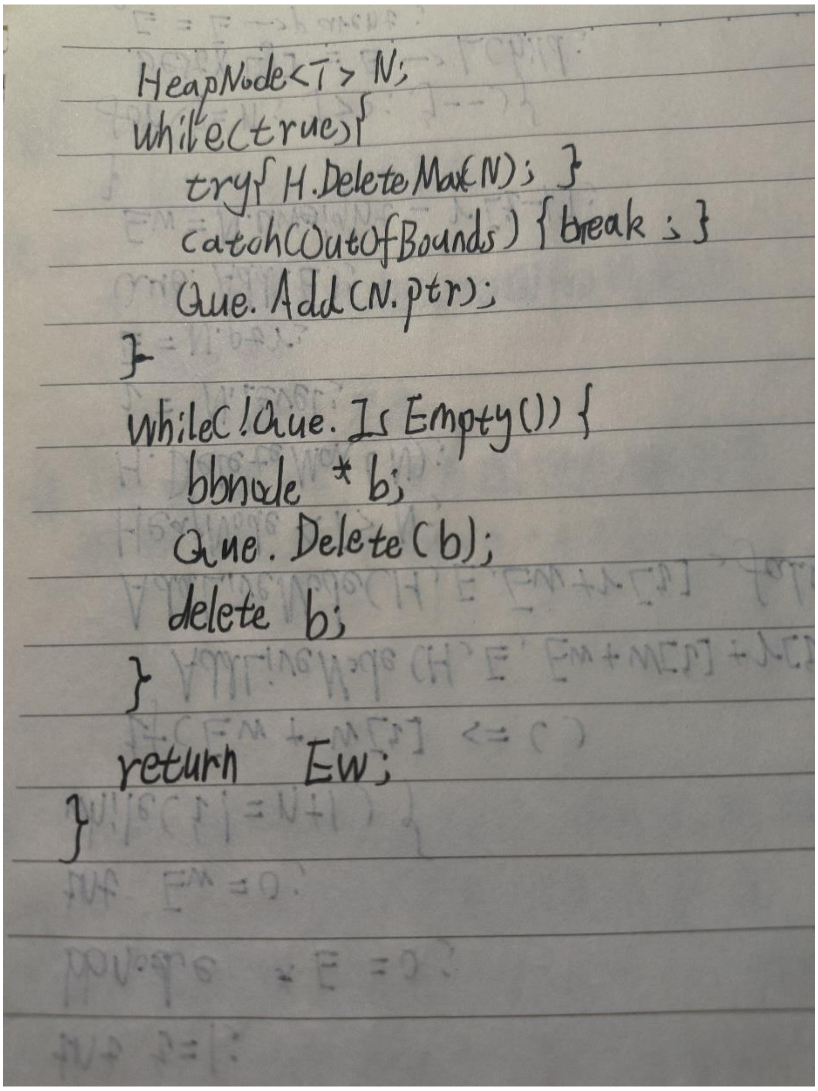
修改后的0-1背包问题的优先队列式分支限界算法如下：

```
template<class Typew, class Typep>
Typep knap<Typew, Typep>::Maxknapsack(){
   H = new MaxHeap<HeapNode<Typep, Typew> >(1000);
  Queue<bbnode *> Que;
  bestx = new int[n+1];
  int i = 1;
  E = 0;
  cw = cp = 0;
  Typep bestp = 0;
  Typep up = Bound(1);
 while(i != n+){
    Typew wt = cw + w[i];
    if(wt <= c){
      if(cp + p[i] > bestp)
          bestp = cp + p[i];
   AddLiveNode(up, cp+p[i], cw+w[i], true, i+1);}
   up = Bound(i+1);
   if(up>=bestp)
      AddLiveNode(up, cp, cw, flase, i+1);
~~ap~~ up = Bound(i+1)
  if(up>=bestp)
    AddLiveNode(up, cp, cw, flase, i+1);
  HeapNode<Typep, Typew> N;
  H->DeleteMax(N);
  E = N.ptr;  Que.Add(E];
  cw=N.weight; cp=N.profit; up=N.uprofit;
  i = N.level;}
 for(int j=n; j>0; j--){
   bestx[j] = E->LChild;
   E = E->parent;
 }
 HeapNode<Typep, Typew>N;
 while (true){
     try{ H->DeleteMax(N);}
     catch (OutOfBounds){break;}
     Que.Add(N.ptr);  }
while(!Que.IsEmpty()){
    bbnode *b;
    Que.Delete(b);
    delete b; }}
return cp;
}
```

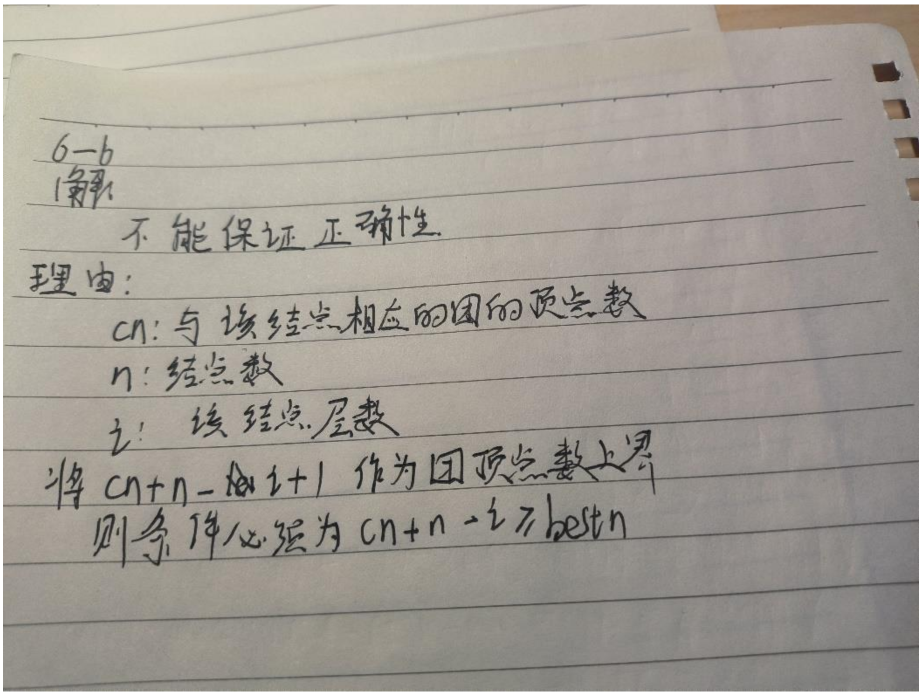


修改后的解装载问题的优先队列式分支如下:

```
template<class T>
T MaxLoading(T*w, T c, int n, int *bestx){
  MaxHeap<HeapNode<T>> H(1000);
  Queue<bbnode*> que;
  T *r = new T[n+1];
  r[n] =0;
  for(int j=n-1; j>0; j--)
    r[j] = r[j+1] + w[j+1];
  int i=1;
  bbnode *E =0;
  int Ew =0;
  while(i!=n+1){
    if(Ew + w[i] <= c)
      AddLiveNode(H, E, Ew+w[i]+r[i], true, i+1);
    AddLiveNode(H, E, Ew+r[i], false, i+1);
    HeapNode<T> N;
    H.DeleteMax(N);
    i = N.level;
    E = N.ptr;
    que.Add(E);
    Ew=N.uweight - r[i-1];
  }
  for(j=n; j>0; j--){
    bestx[j] = E ->Lchild;
    E = E->parent;
  }
}
```



```
HeapNode<T> N;
while(true){
    try{ H.DeleteMax(N); }
    catch(OutOfBounds){break ;}
    Que. Add(N.ptr);
}
while(!Que. Is Empty()){
    bbnode *b;
    Que. Delete(b);
    delete b;
}
return Ew;
}
```



6-b

解:

不能保证正确性

理由:

cn: 与该结点相应的团的顶点数

n: 结点数

i: 该结点层数

将 cn+n-i+1 作为团顶点数上界

则条件必须为 cn+n-i>=bestn