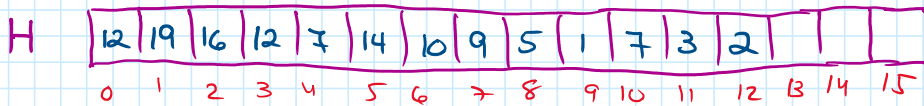


i צומח
 $2i$ בן שמאל
 $2i+1$ בן ימין
 $i/2$ אבא

ערימה : Heap

אלד הקטנה שיהי MAX



נבחר הערך

שתי פעולות עזר

- 1. $BubbleUp(H, i)$ - מעלה כלפי מעלה את הערך באינדקס i .
 - 2. $SiftDown(H, i)$ - מוריד כלפי מטה את הערך באינדקס i .
- באינדקס i נמצא ערך שמספר אביו גדול מהערך

```

BubbleUp (int H[], int i)
{
    while ( i > 1 && H[i] > H[i/2] )
    {
        swap ( H[i], H[i/2] );
        i = i/2 ;
    }
}
    
```

מספר ההחלפה הוא א
מחצית 4 פעולות
במקרה האולטימלי

```

BubbleUp (int H[], int i)
{
    temp = H[i];
    while (i > 1 && temp > H[i/2])
    {
        H[i] = H[i/2];
        i = i/2 ;
    }
    H[i] = temp;
}
    
```

מספר ההחלפה הוא א
מחצית 2+2 פעולות
2 פעולות האולטימלי + העלה

$SiftDown (int H[], int i)$

מספר אבא שלי הולך

SiftDown (int H[], int i)

④

```
int temp = H[i];
```

```
while ( H[0] > i*2 ) // עד ג'ט
```

largest = i * 2; // הכפול הגדול ביותר

if $(H[0] \neq i * 2 + 1)$ // 'n' is prime

if ($H[i*2+1] > H[largest]$) $largest = i*2+1$;

היבן הימני יבני גבול

```
if (temp > H[largest]) { H[i] = temp ; break ; }
```

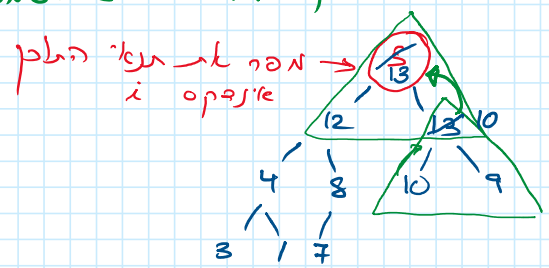
largest האנרגיה של היצור מבין האנרגיה והזמן

$$H[i] = H[\text{largest}];$$

```
i = largest;
```

3

```
H[i] = temp;
```



$i = 13$ $\text{temp} = 5$
 0 1 2 3 4 5 6 7 8 9 10
 10 | 5 12 13 4 8 10 9 3 1 7
 4 4 4
 largest $\times 3$
 $\neq 6$
 $3 =$
 $4 \times 2 + 1$

```
Insert (int t[], int newnum)
```

④

משמרים נאמן //

$$H[0]++;$$
$$H[H[0]] = \text{new num};$$

מכנים את המספר החזק
עם המילה ומעבירים אותו

משמרים לבן

BubbleUp ($H, H[0]$);



פה יושב היצחק
החדש שלי. מפה
את תנאי הילדן

```
int DeleteMax (int H[])
```

2

$$\max = H[1];$$

```

H[1] = H[H[0]]; // נבחר את השורש עם העלה האחרון
H[0]--; // נציב את מקום הענף
SiftDown(H, 1); // נשאר את תנאי התלכך
return max;

```

]