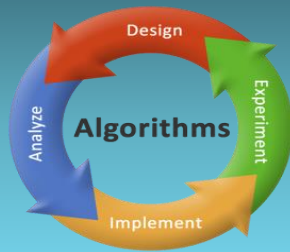


# الگوریتم‌های حریم‌سازی

کد هافمن

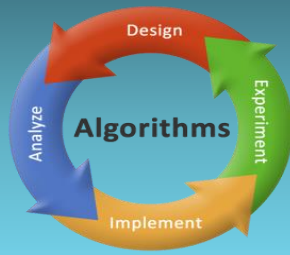
# کد هافمن



❑ اکثراً برای ذخیره فایل‌های متنی از فرمت‌های ASCII یا UTF8 استفاده می‌شود. این فرمت‌های استاندارد برای ذخیره‌ی هر کاراکتر از یک کد ۸ بیتی استفاده می‌کنند.

❑ کد هافمن روشی ارائه می‌دهد تا برای هر کاراکتر به نسبت تکرار آن در متن کدی با طولی خاص استفاده شود؛ یعنی برای کاراکترهایی با فراوانی بالاتر در متن، کدهایی کوتاه‌تر استفاده شود.

# مثال کد هافمن



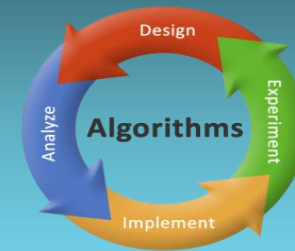
❑ فرض کنید متنی دارای ۱۰۰,۰۰۰ کاراکتر داریم که فقط از کاراکترهای abcdef تشکیل شده است. اگر بخواهیم از کد با طول ثابت استفاده کنیم به ۳ بیت برای هر کاراکتر نیاز داریم؛ یعنی ۳۰۰,۰۰۰ بیت و برای استفاده کد ASCII ۸۰۰,۰۰۰ بیت.

❑ حال اگر از کد هافمن استفاده کنیم:

	a	b	c	d	e	f
فراوانی در متن %	45	13	12	16	9	5
کد هافمن	0	101	100	111	1101	1100

$$(45 * 1 + 13 * 3 + 13 * 3 + 16 * 3 + 9 * 4 + 5 * 4) * 1000 = 224000$$

# کدهای پیشوندی



کاراکترها	a	b	c	d	e	f
فراوانی در متن. %	45	13	12	16	9	5
کد ما	0	10	01	1	00	11
کد هافمن	0	101	100	111	1101	1100

☐ عبارت abbd را در نظر بگیرید.

☐ رمز شده آن با کد هافمن خواهد بود: 0101101111

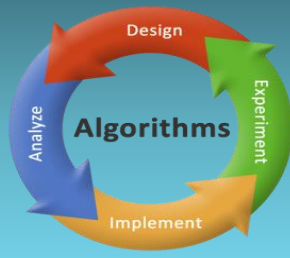
☐ با کد ما: 010101

☐ بر اساس کد ما این کد معادل چه عبارتی است؟

☐ بر اساس کد هافمن ؟ 0/101/101/111:abbd

01/01/01	ccc
0/10/1/01	abdc
0/1/0/1/0/1	adadad
0/10/10/1	abbd

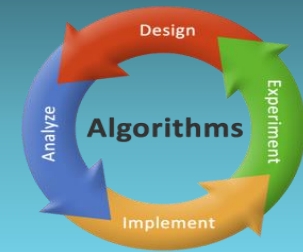
# درخت هافمن



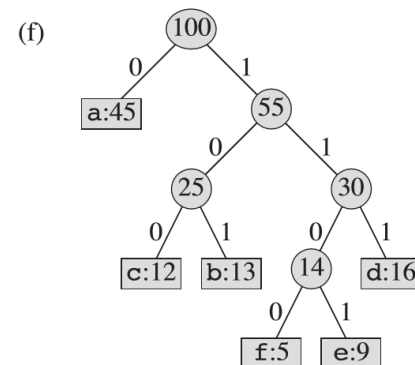
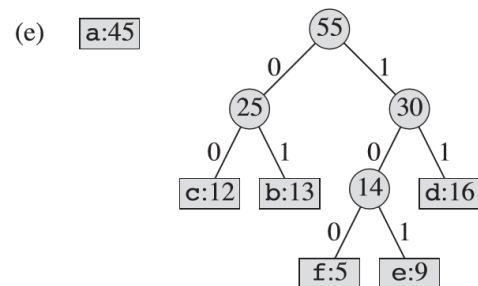
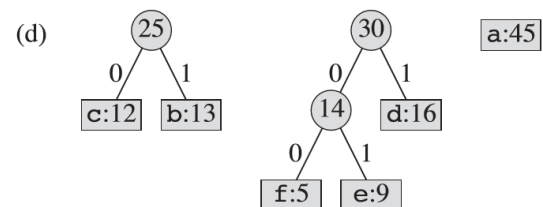
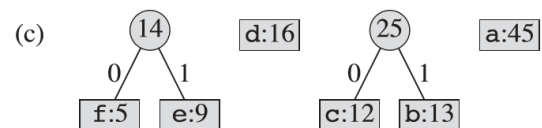
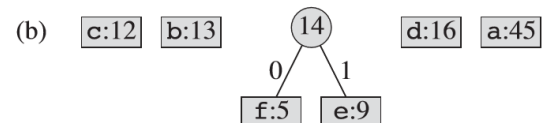
HUFFMAN( $C$ )

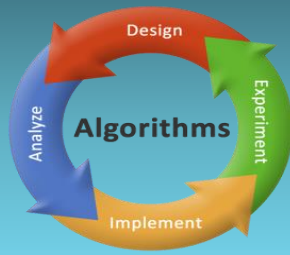
```
1   $n = |C|$ 
2   $Q = C$ 
3  for  $i = 1$  to  $n - 1$ 
4      allocate a new node  $z$ 
5       $z.left = x = \text{EXTRACT-MIN}(Q)$ 
6       $z.right = y = \text{EXTRACT-MIN}(Q)$ 
7       $z.freq = x.freq + y.freq$ 
8       $\text{INSERT}(Q, z)$ 
9  return  $\text{EXTRACT-MIN}(Q)$     // return the root of the tree
```

# درخت هافمن



(a) f:5 e:9 c:12 b:13 d:16 a:45





# مرور مراحل استفاده از کد هافمن

## □ کد گذاری

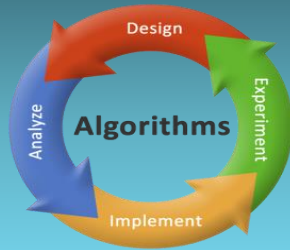
- ✓ ۱- اسکن کردن متن و بدست آوردن فراوانی هر کاراکتر
- ✓ ۲- ساخت درخت هافمن
- ✓ ۳- اسکن دوباره متن و جایگذاری کدهای بدست آمده از درخت

## □ تبدیل دوباره کد به متن

- ✓ ۱- باید درخت هافمن یا جدول کدها را داشته باشیم
- ✓ ۲- با توجه به هر بیت در متن گذشته به طرف راست یا چپ در درخت پیمایش می‌کنیم تا به یک برگ برسیم و سپس کاراکتر بدست آمده را جایگزین می‌کنیم (یا از جدول استفاده می‌کنیم).

# ساخت درخت هافمن

## اسکن متن



*Eerie eyes seen near lake.*

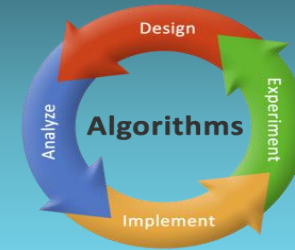
□ دارای چه کاراکترهایی است؟

E e r i space  
y s n a r l k .



# ساخت درخت هافمن

## اسکن متن

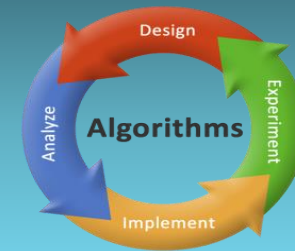


Eerie eyes seen near lake.

□ فراوانی کاراکترها چقدر است؟

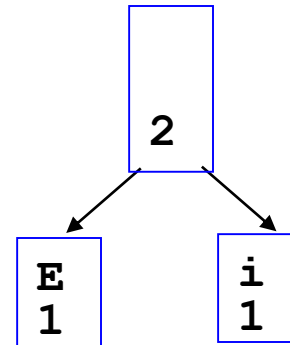
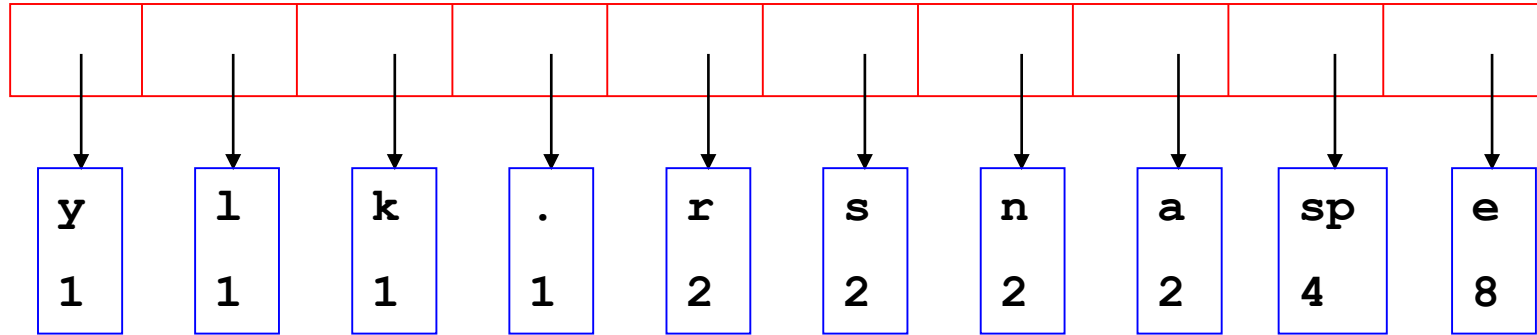
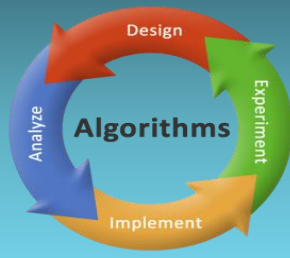
Char	Freq.	Char	Freq.	Char	Freq.
E	1	y	1	k	1
r	2	n	2	.	1
e	2	s	2		
i	1	a	2		
space	4	l	1		

# ساخت درخت هافمن

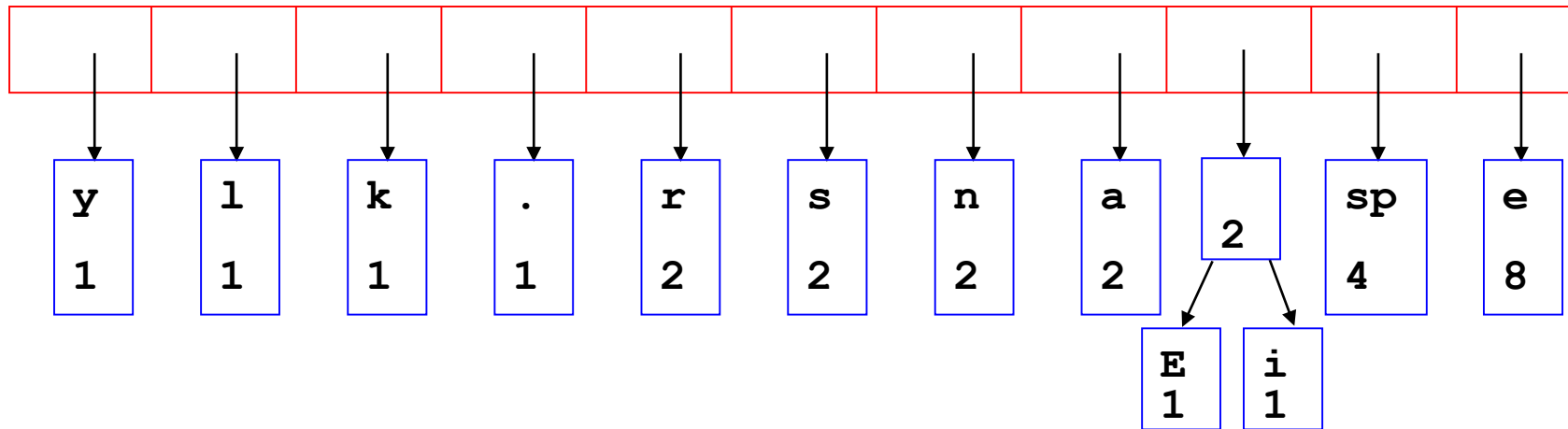
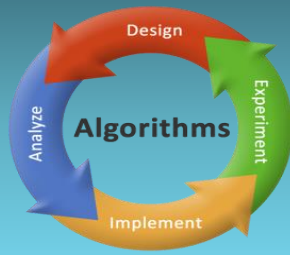


↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
E	i	y	l	k	.	r	s	n	a	sp	e
1	1	1	1	1	1	2	2	2	2	4	8

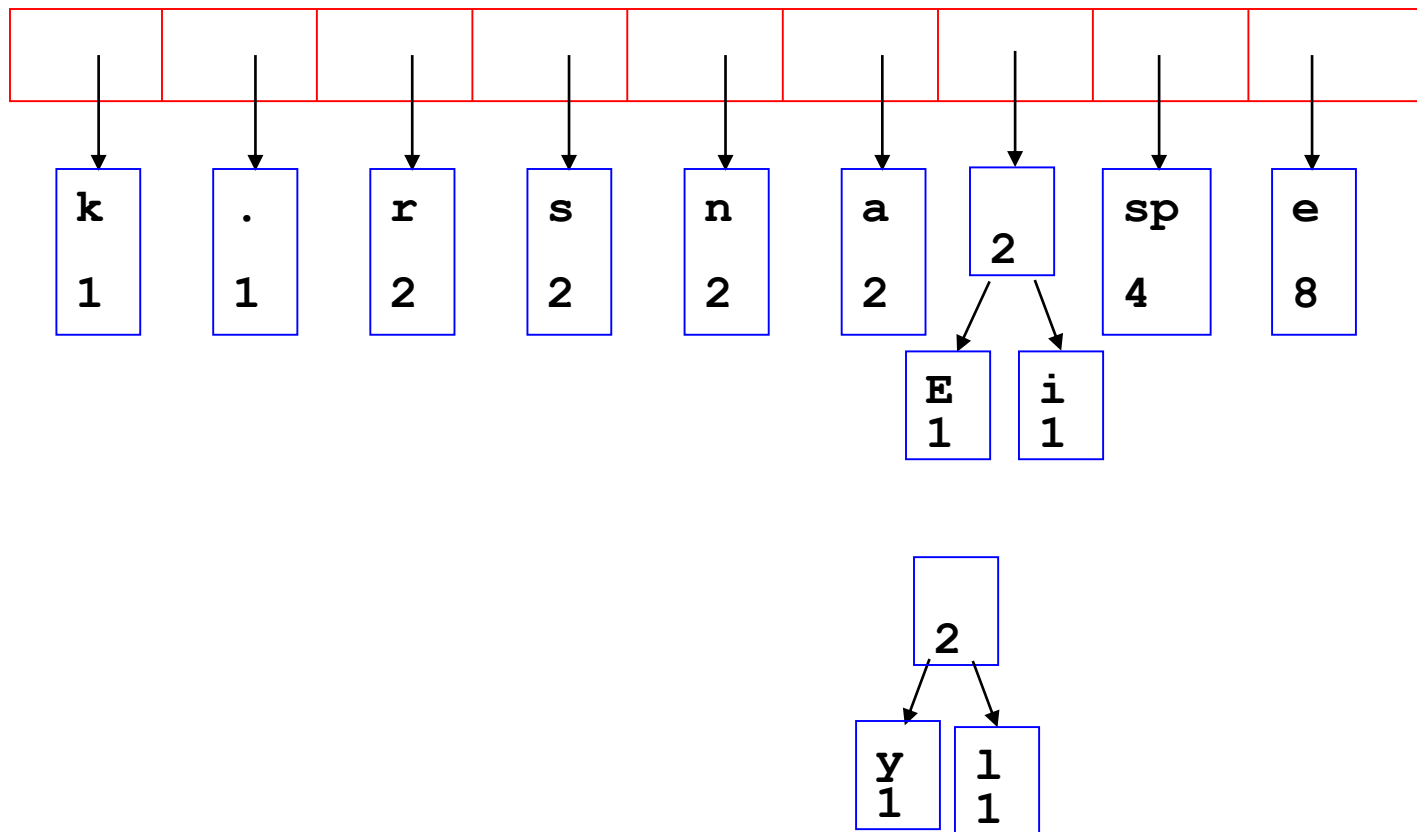
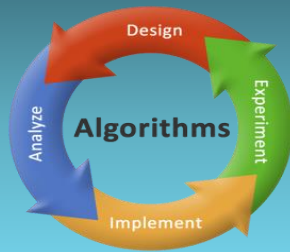
# ساخت درخت هافمن



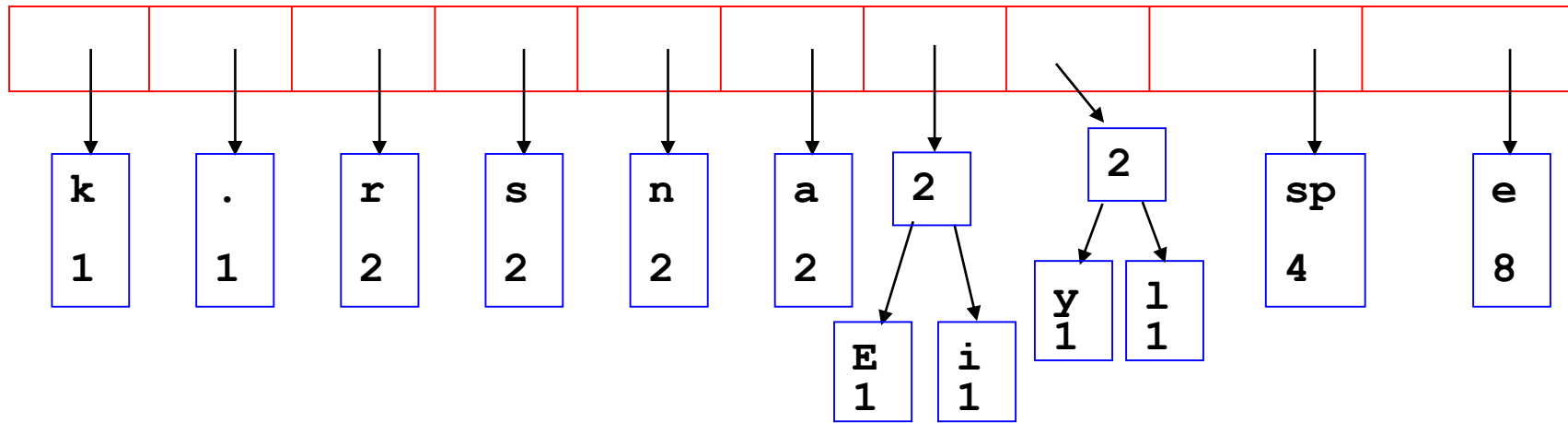
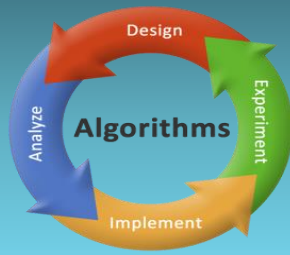
# ساخت درخت هافمن



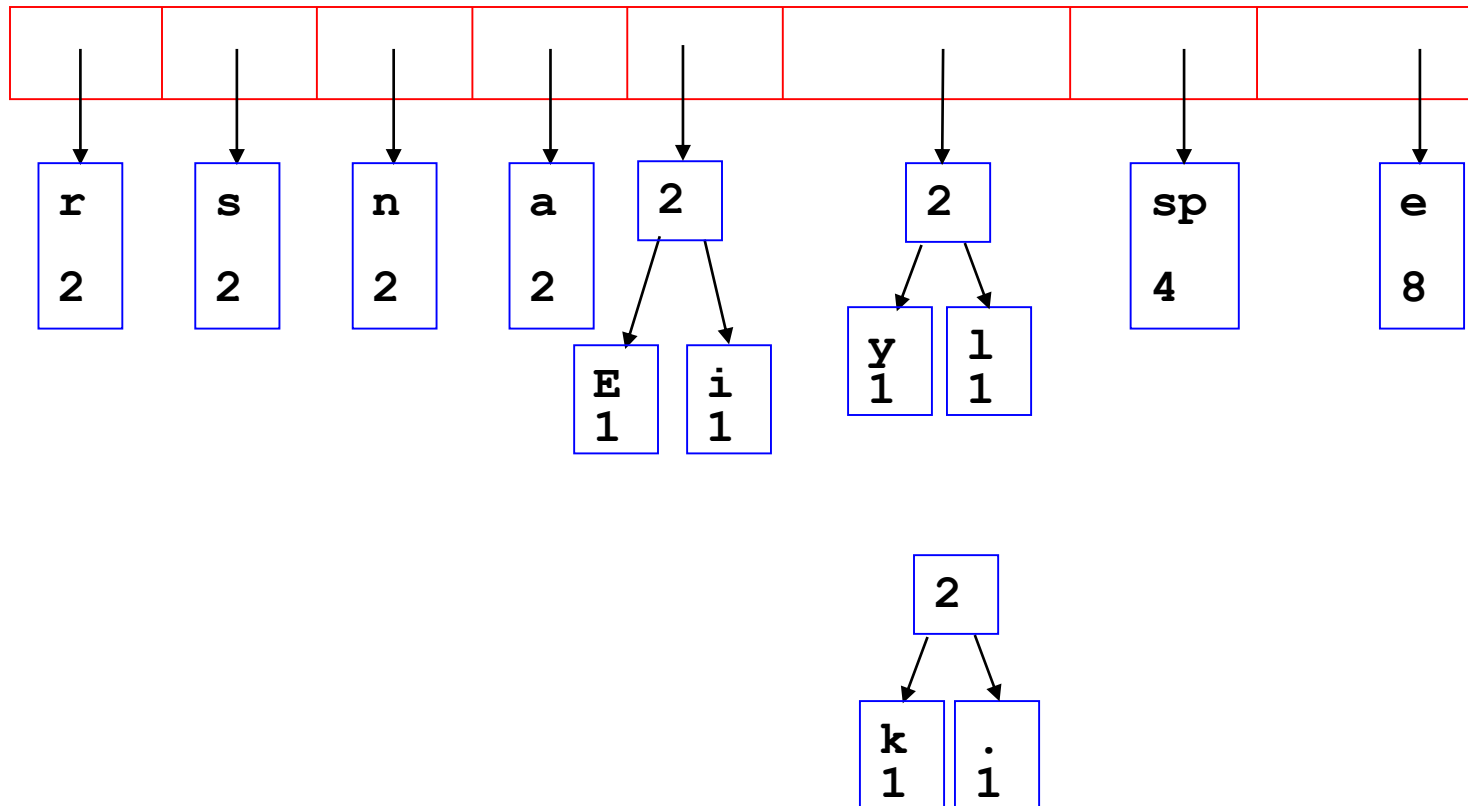
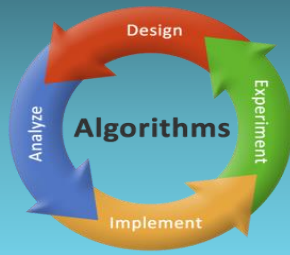
# ساخت درخت هافمن



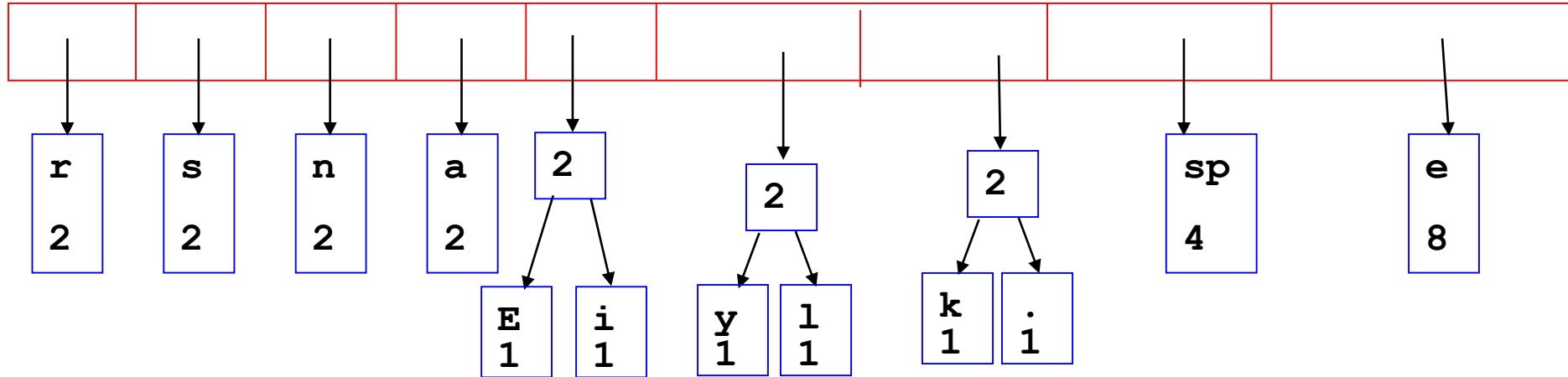
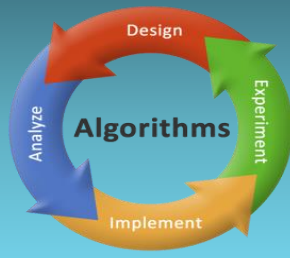
# ساخت درخت هافمن



# ساخت درخت هافمن

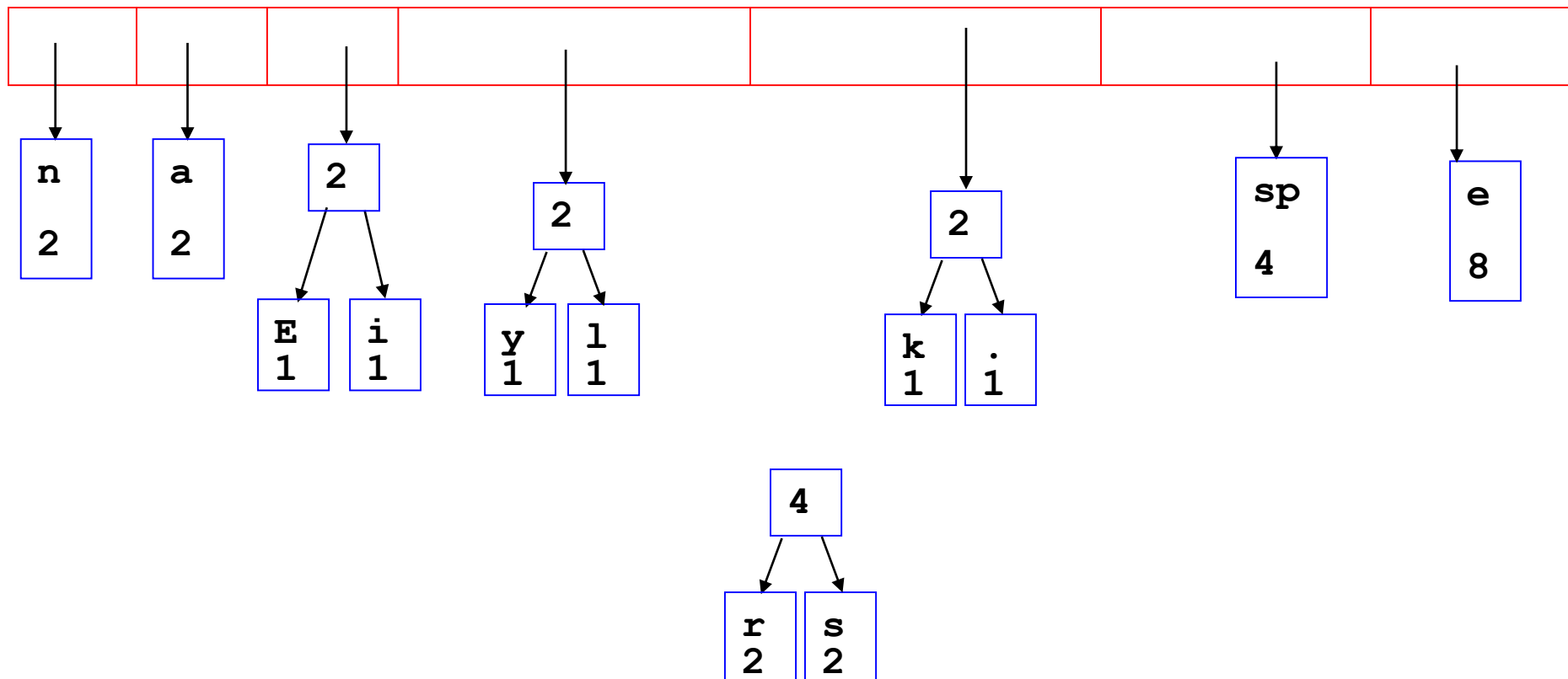
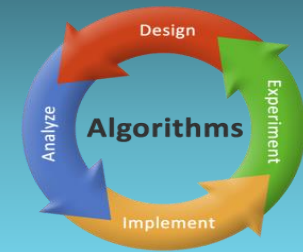


# ساخت درخت هافمن

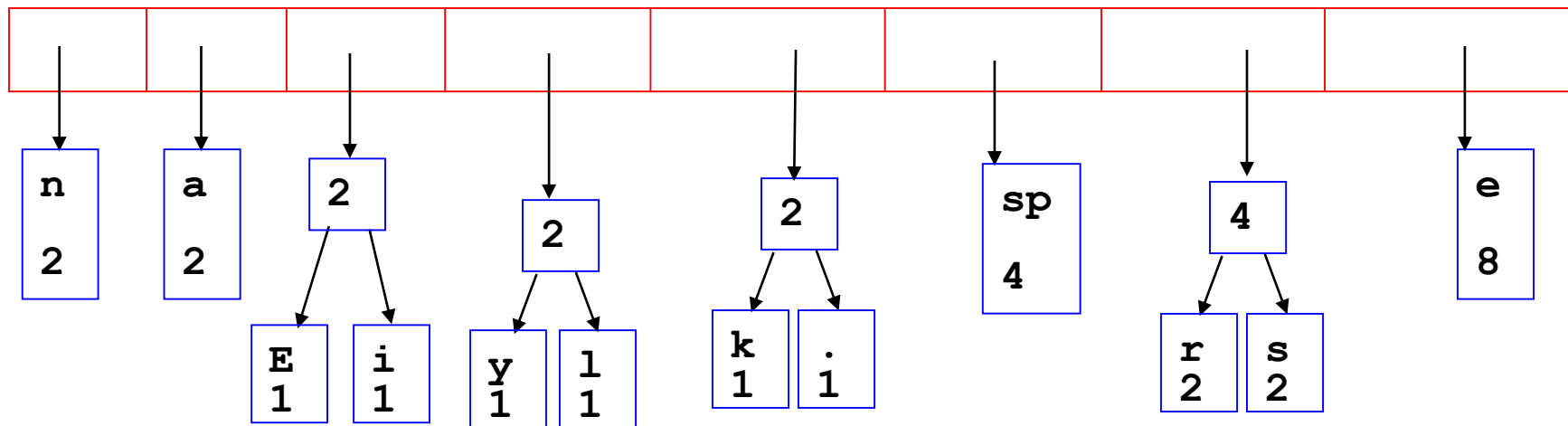
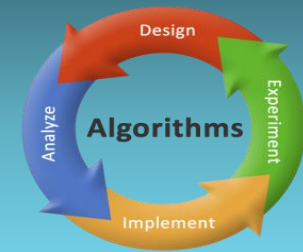




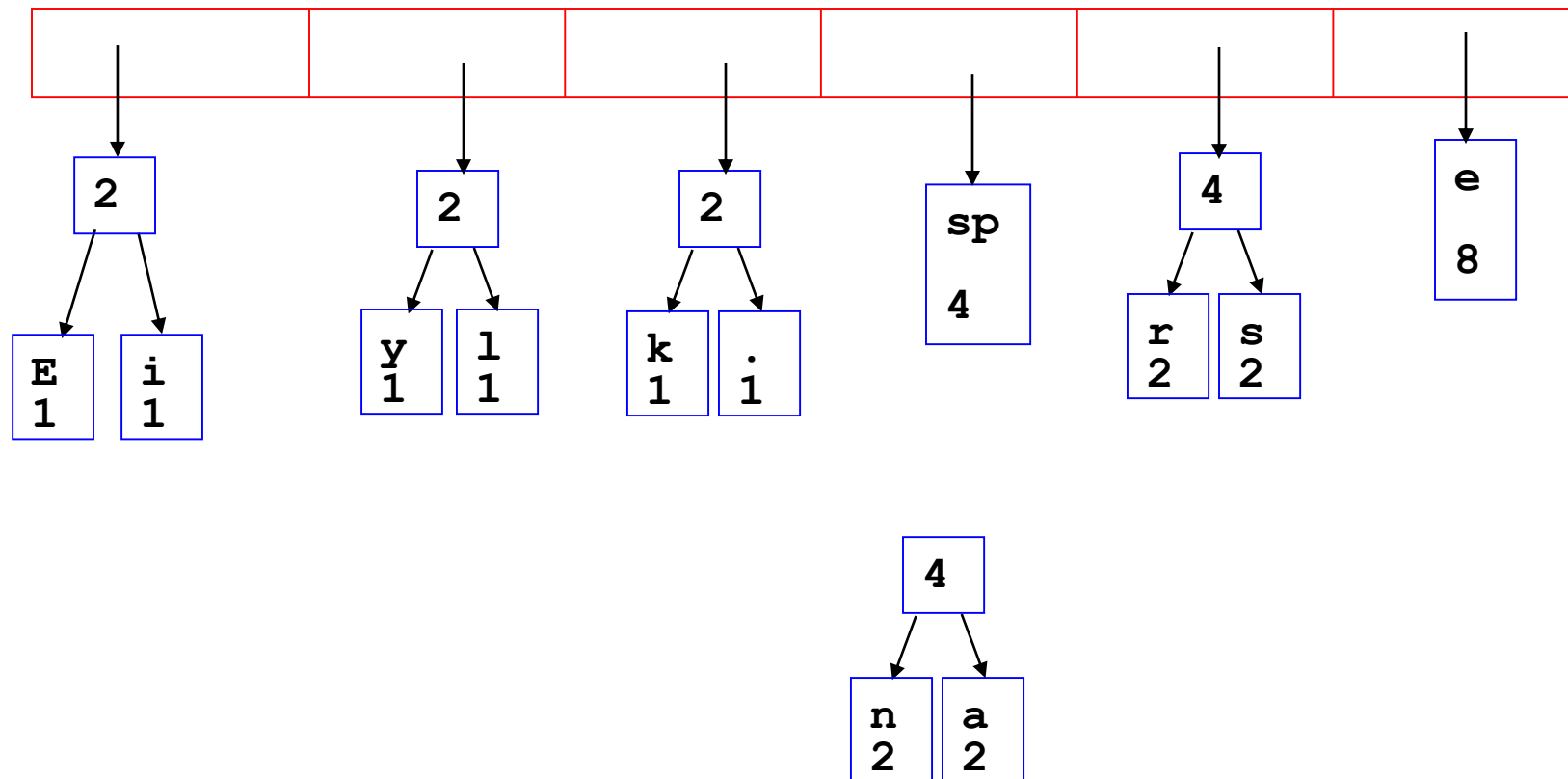
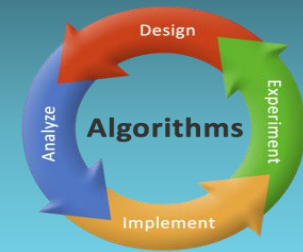
# ساخت درخت هافمن



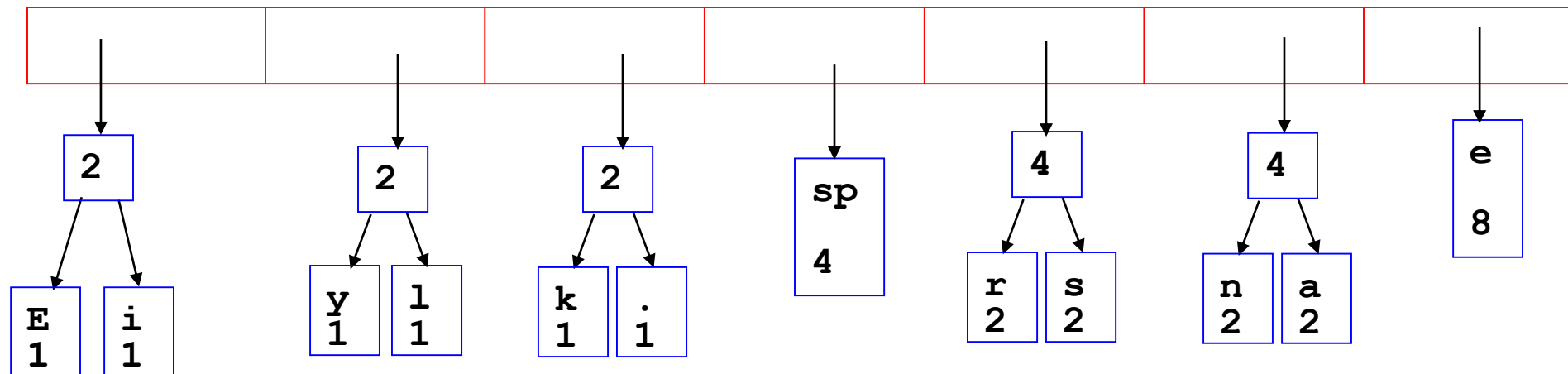
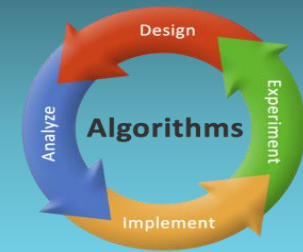
# ساخت درخت هافمن



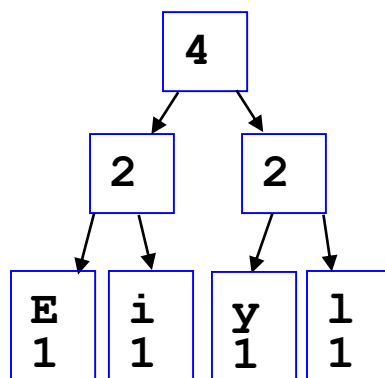
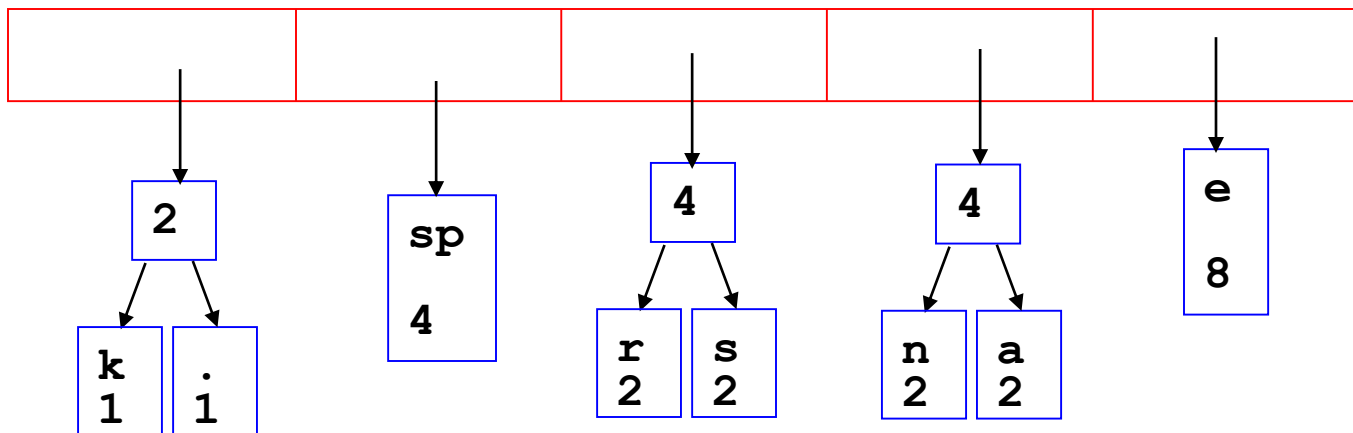
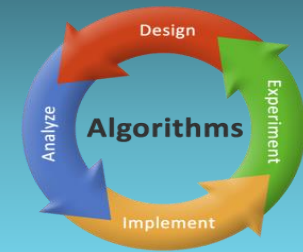
# ساخت درخت هافمن



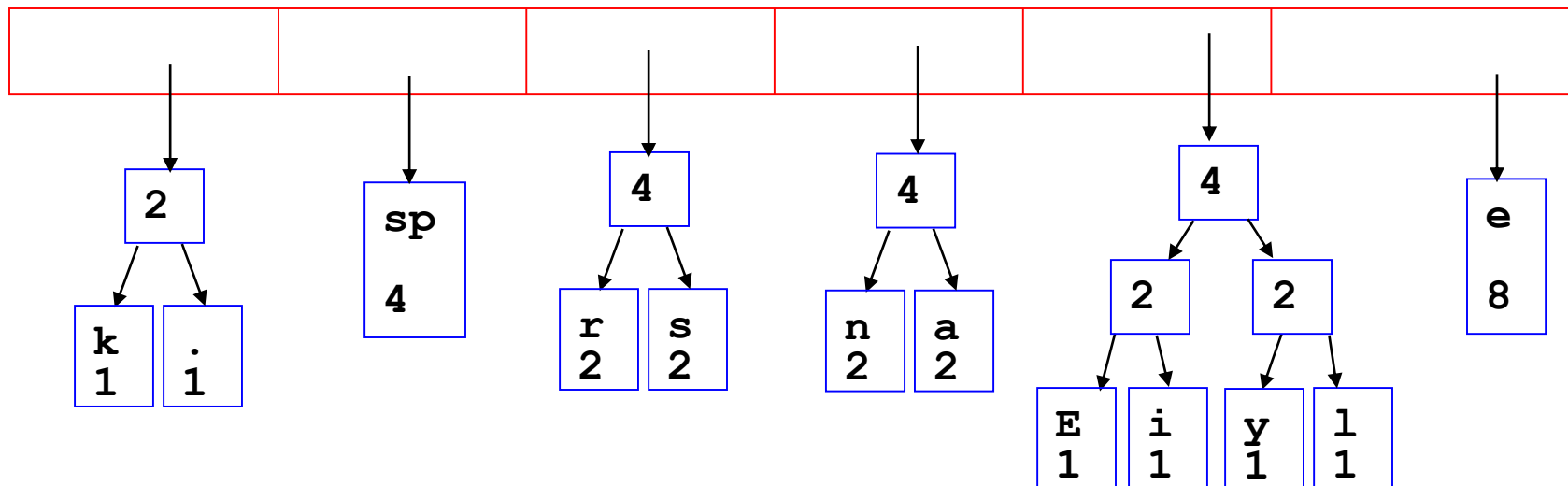
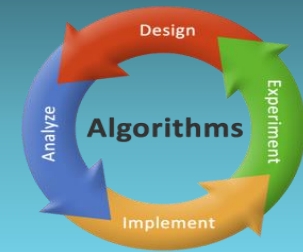
# ساخت درخت هافمن



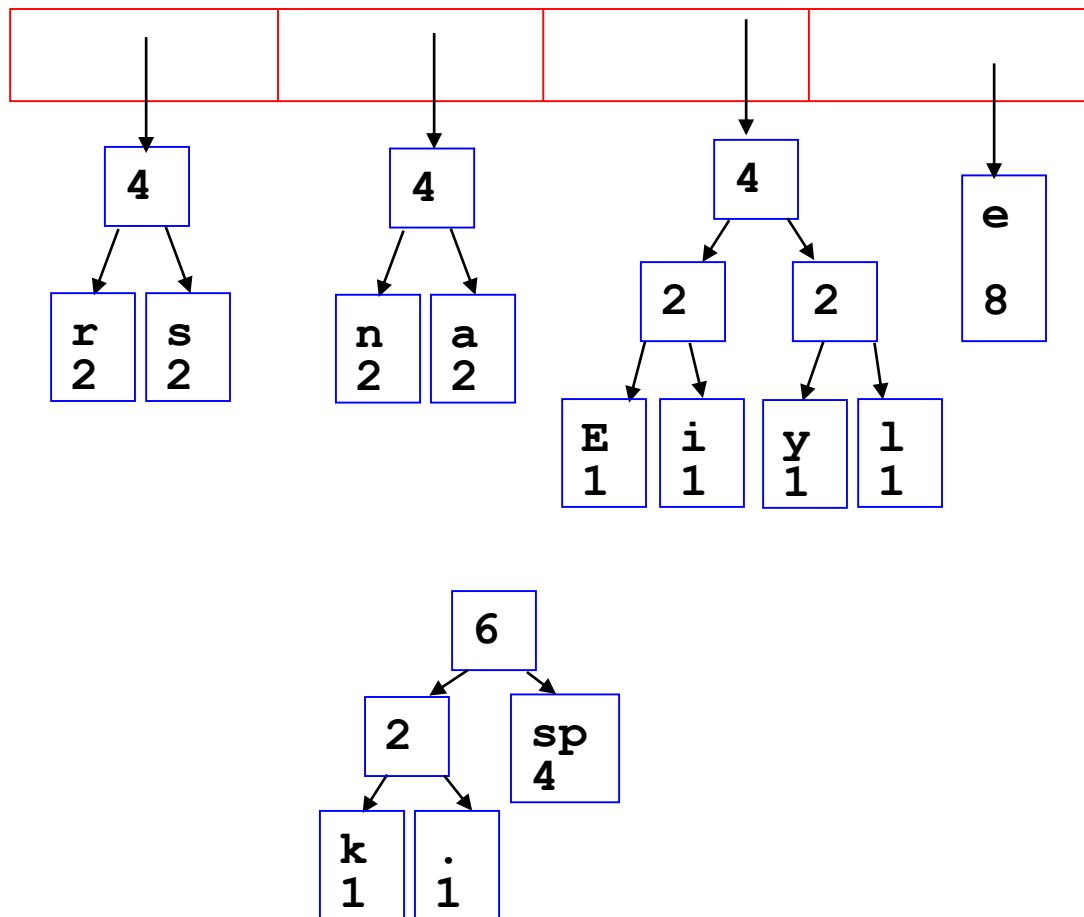
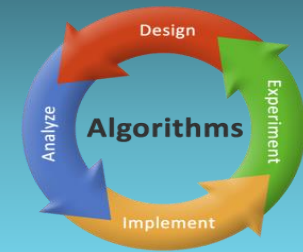
# ساخت درخت هافمن



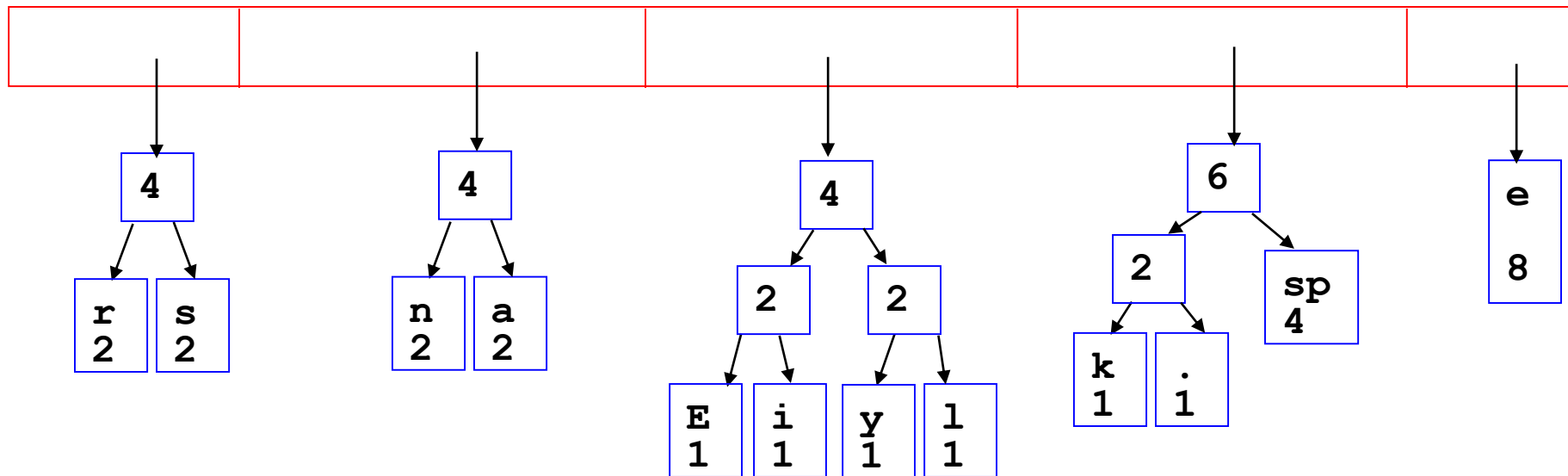
# ساخت درخت هافمن



# ساخت درخت هافمن



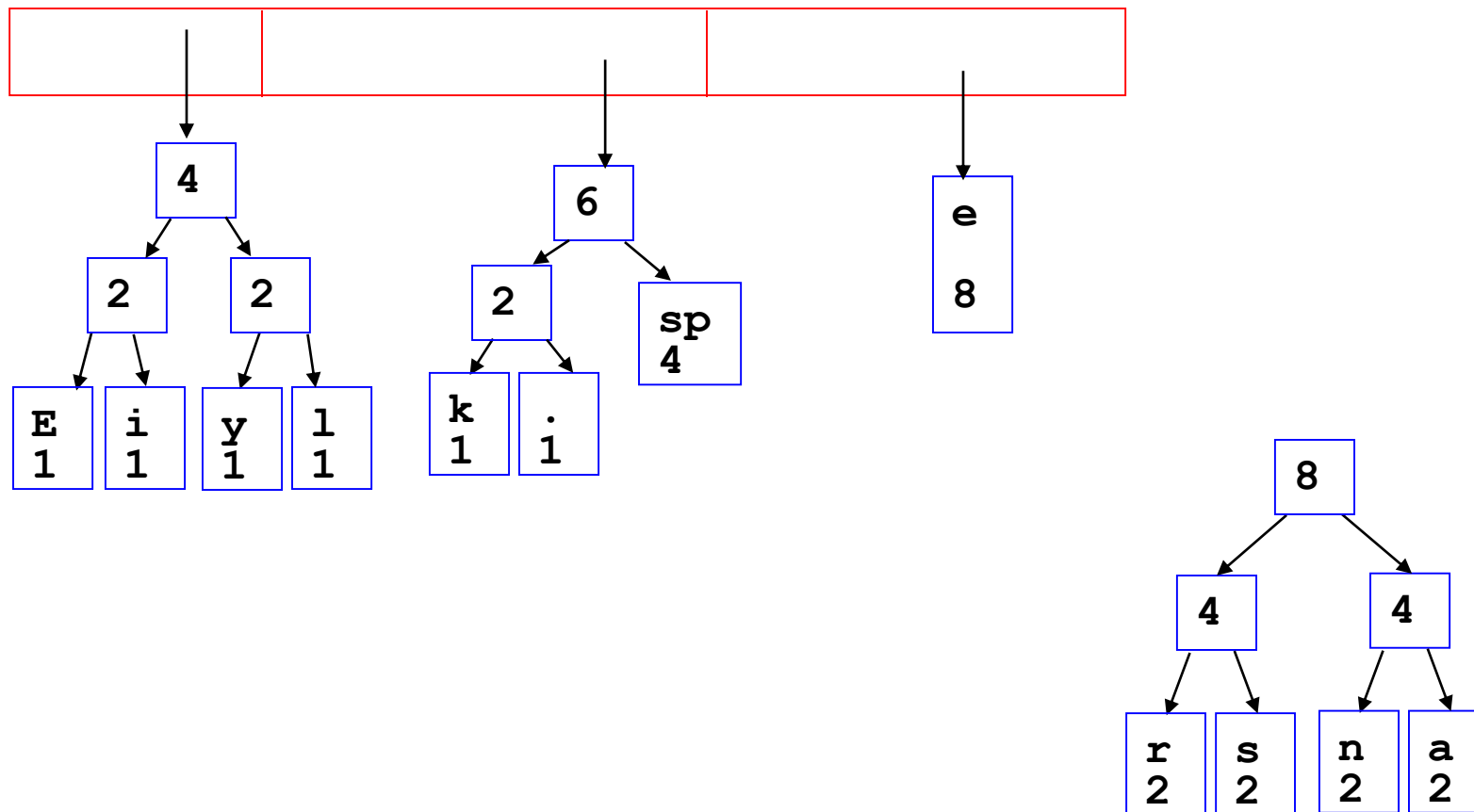
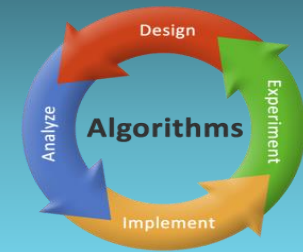
# ساخت درخت هافمن



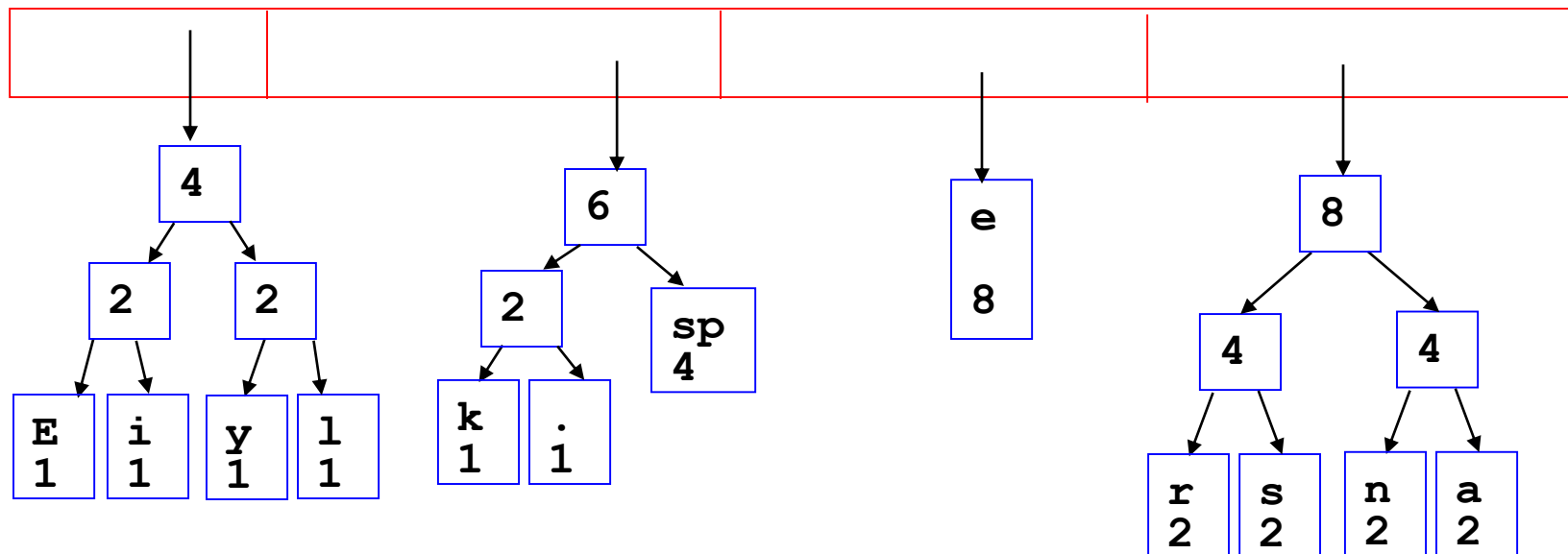
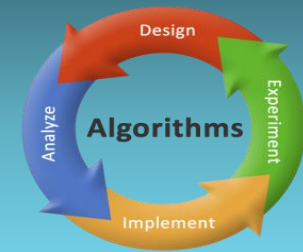
برای کاراکترهایی با فراوانی کم چه اتفاقی در حال وقوع است؟



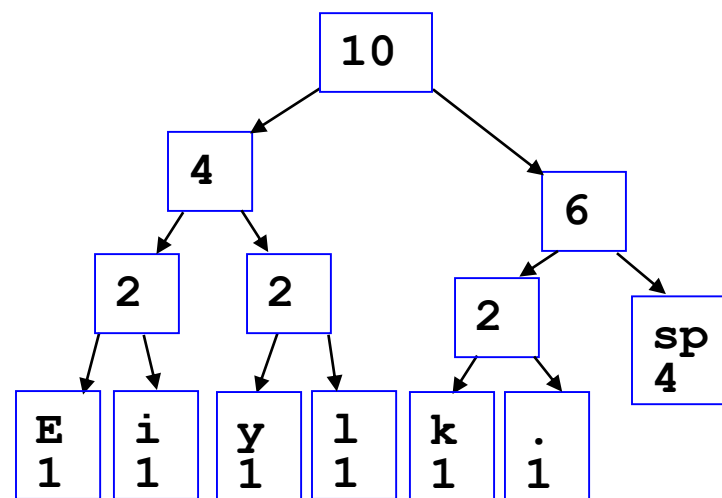
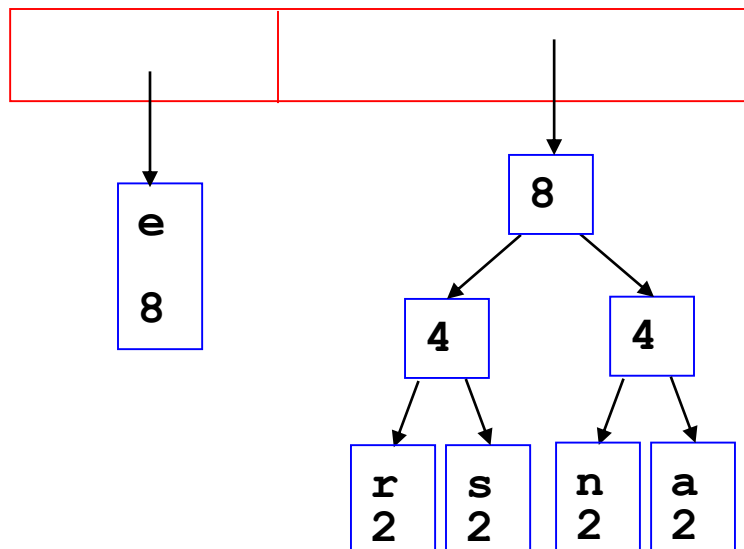
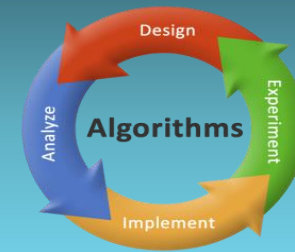
# ساخت درخت هافمن



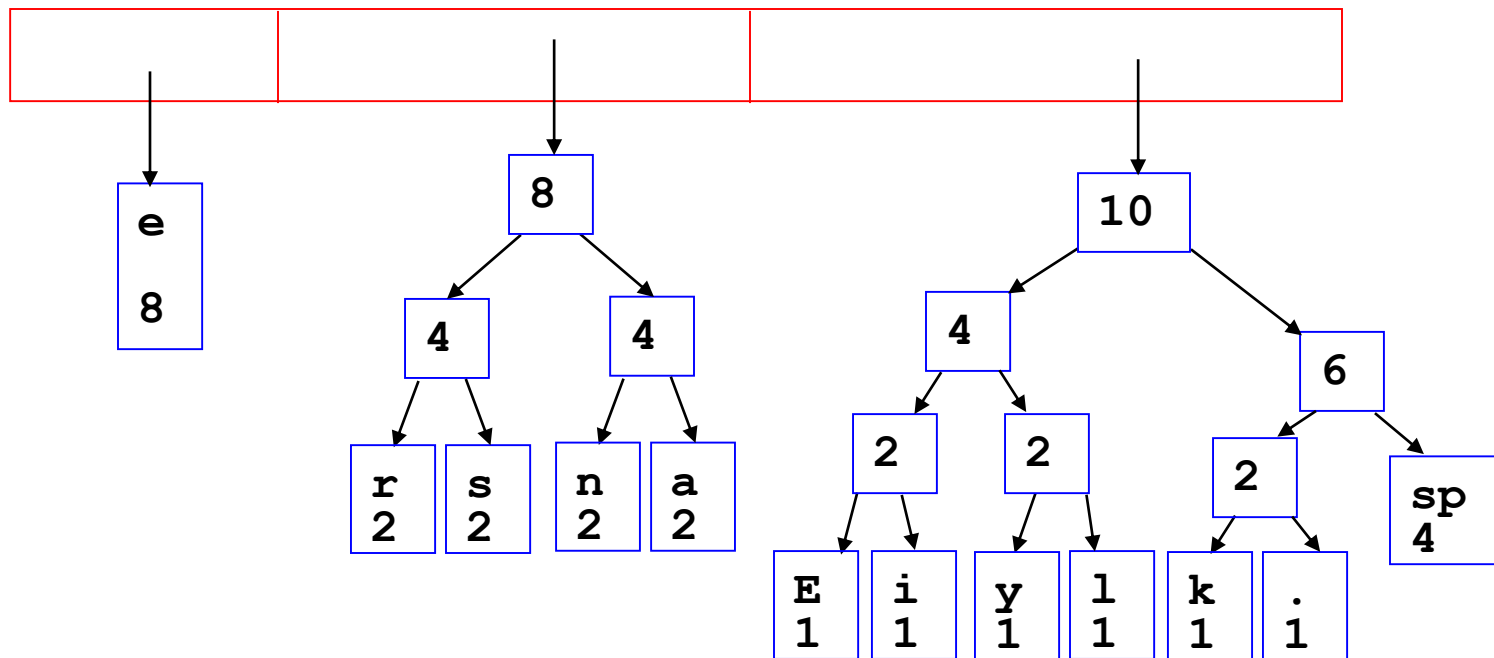
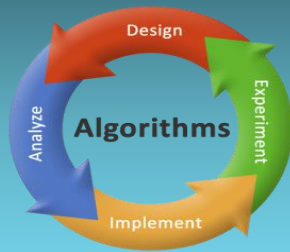
# ساخت درخت هافمن



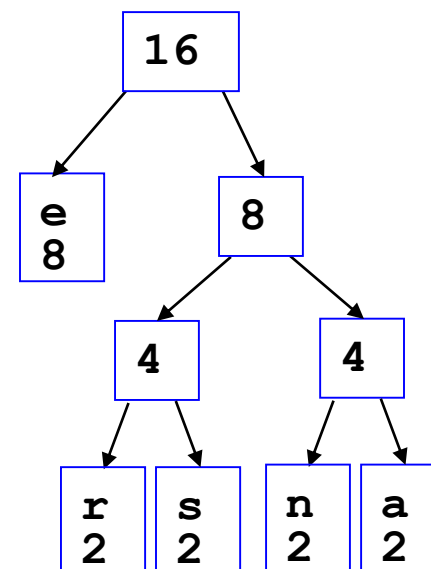
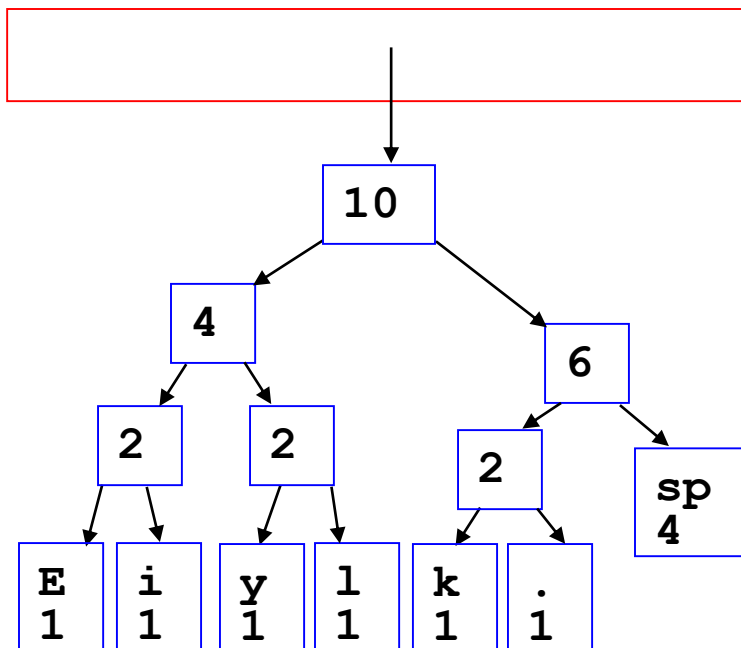
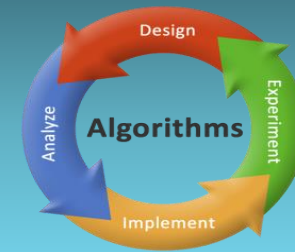
# ساخت درخت هافمن



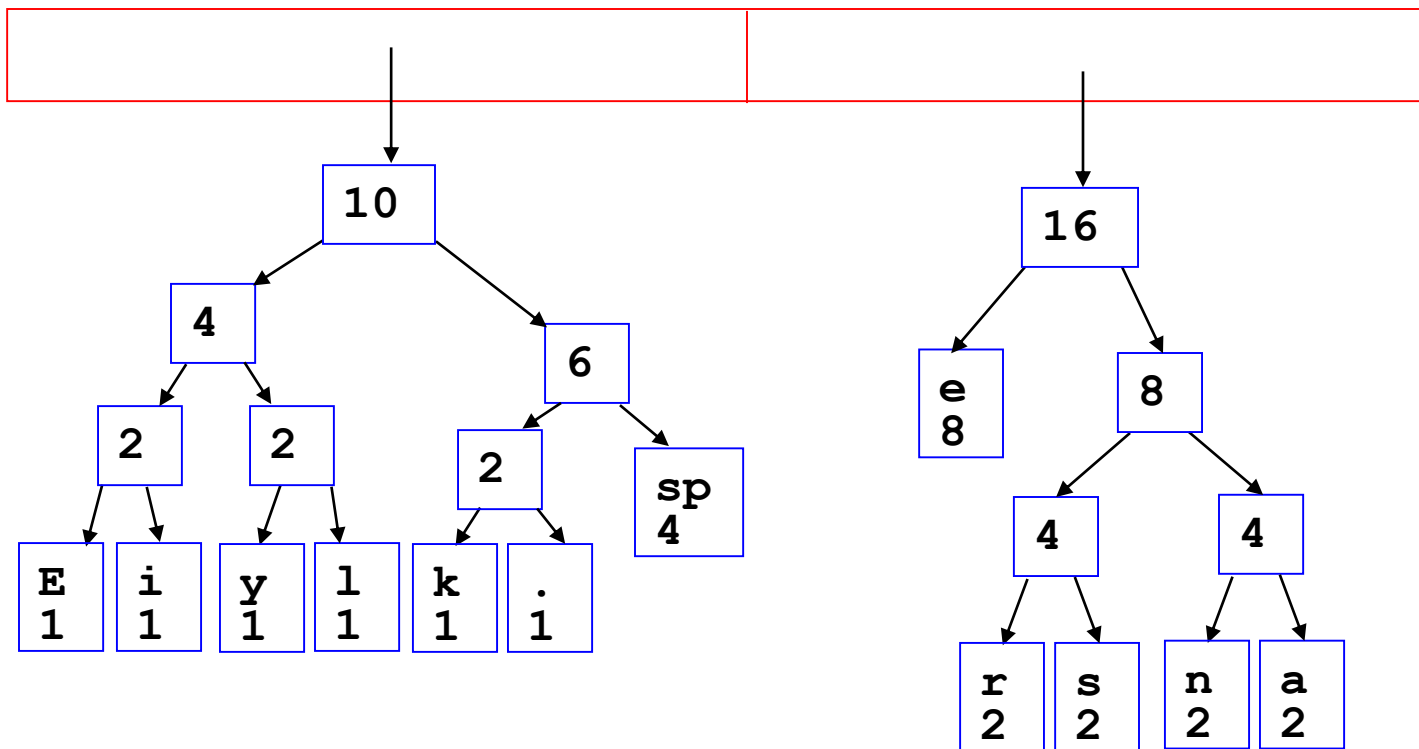
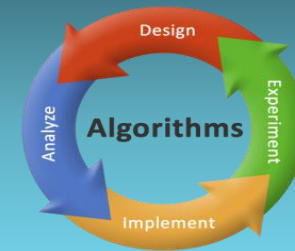
# ساخت درخت هافمن



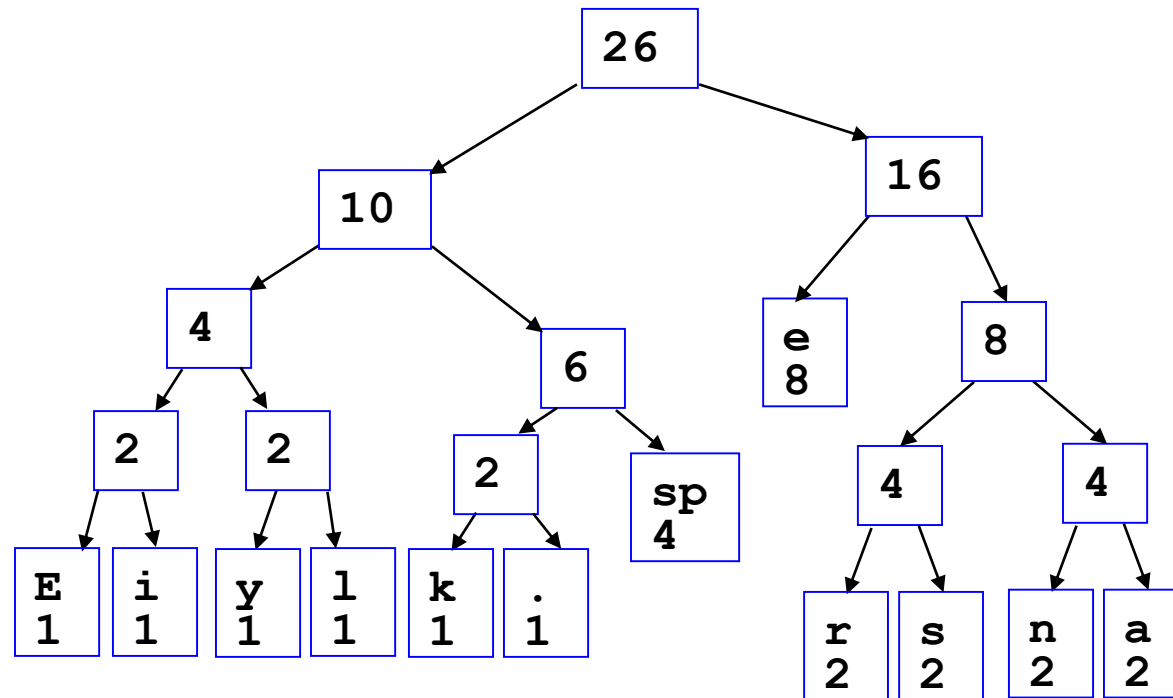
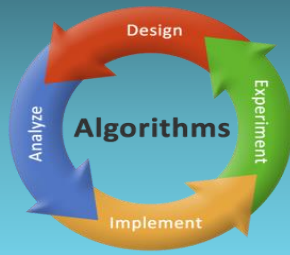
# ساخت درخت هافمن



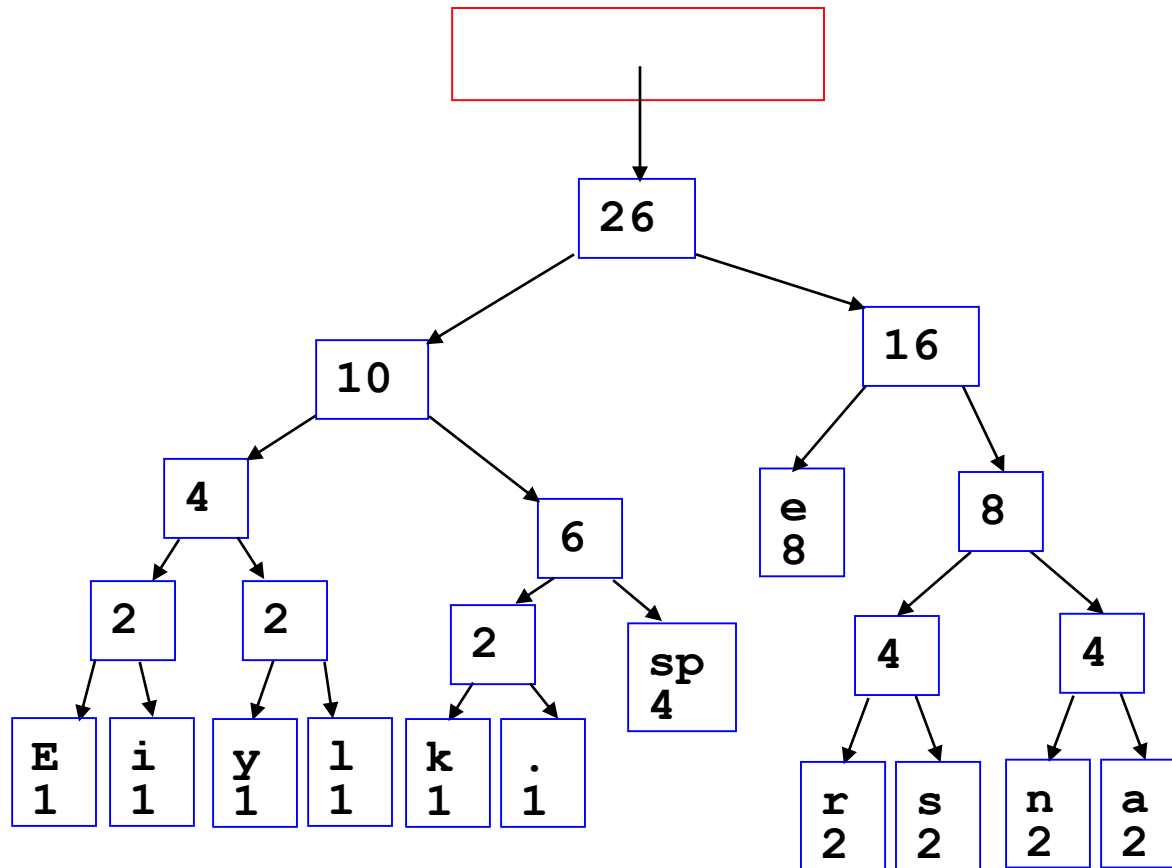
# ساخت درخت هافمن



# ساخت درخت هافمن



# ساخت درخت هافمن

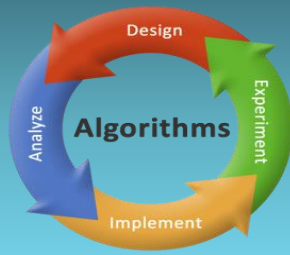


بعد از خروج این نود از صف اولویت، تنها یک نود در صف باقی مانده است که همان ریشه درخت هافمن نهایی است.



# رمز کردن جمله (فایل)

پیمایش درخت هافمن



Code

Char

0000

E

0001

i

0010

y

0011

l

0100

k

0101

.

011

space

10

e

1100

r

1101

s

1110

n

1111

a

