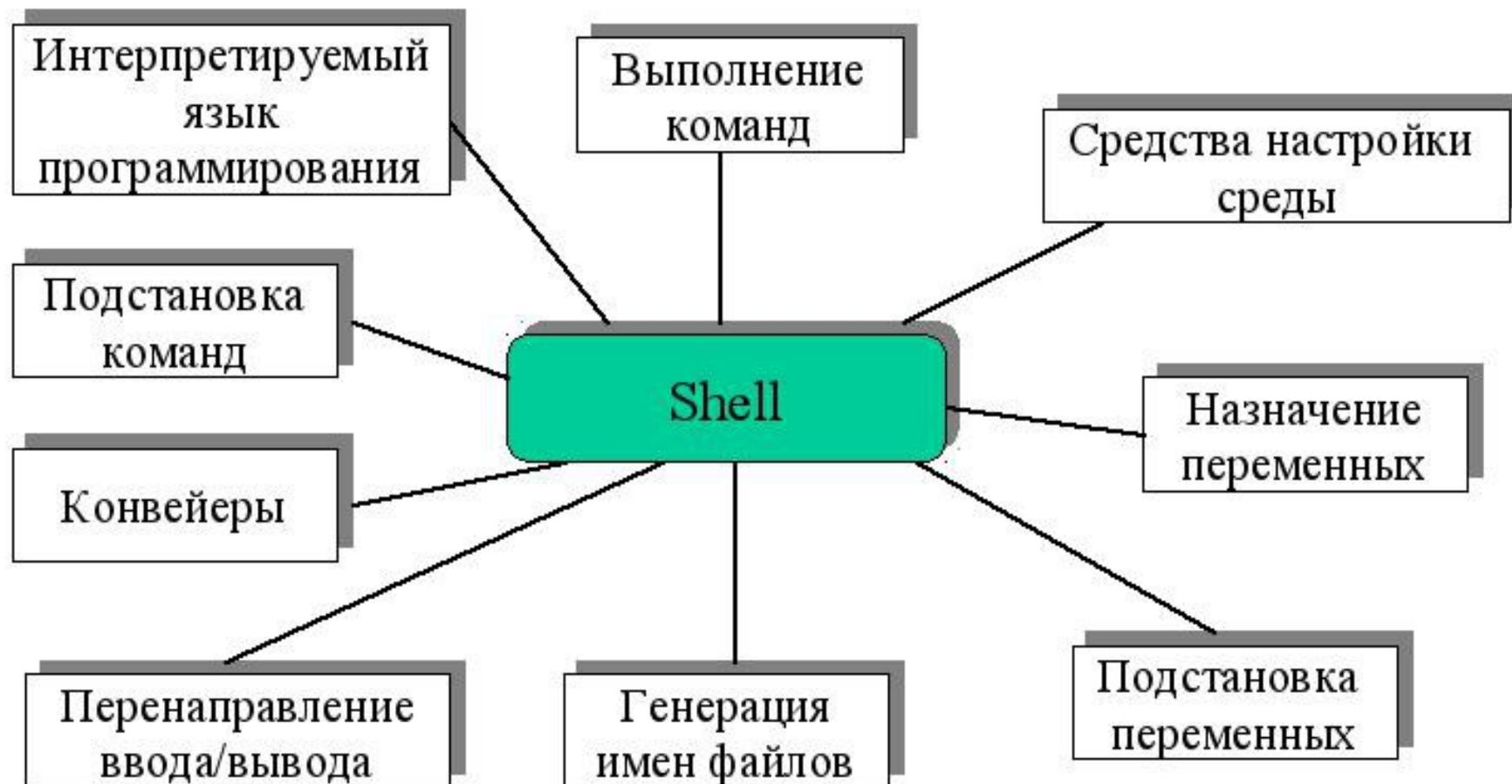
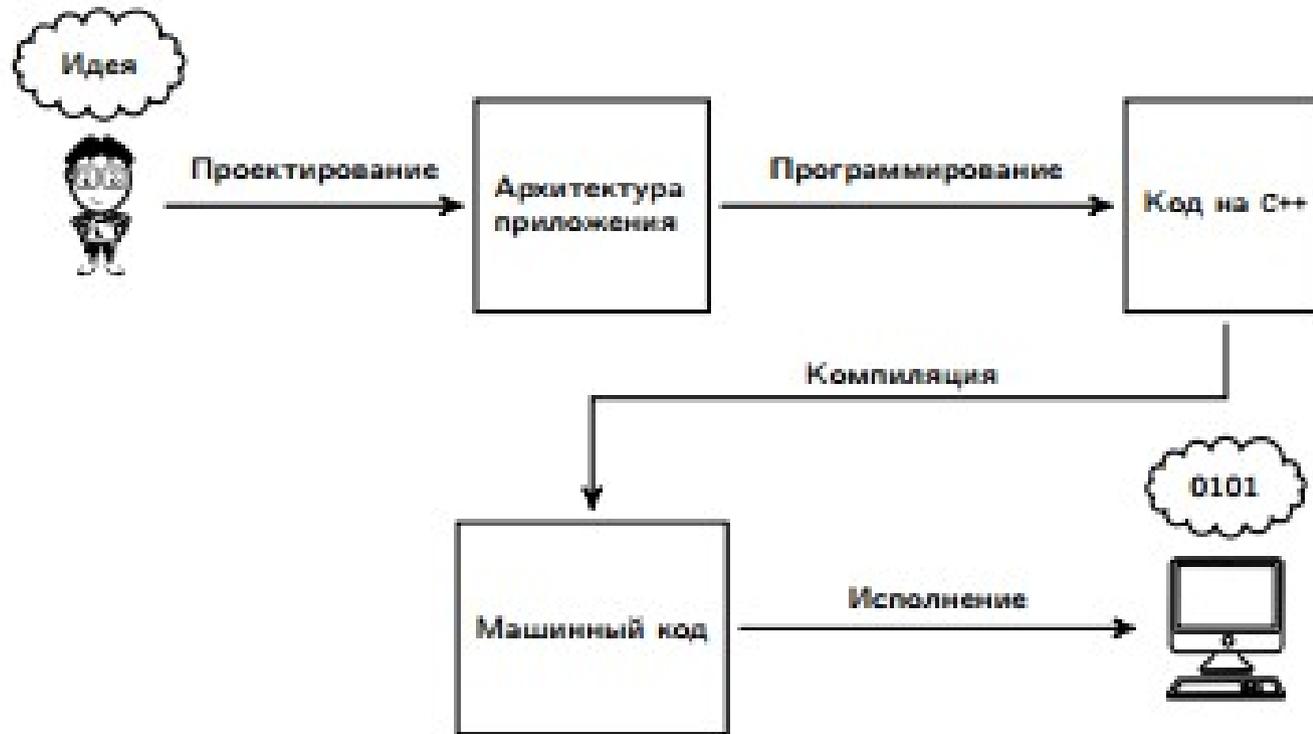


# Bach Scripting

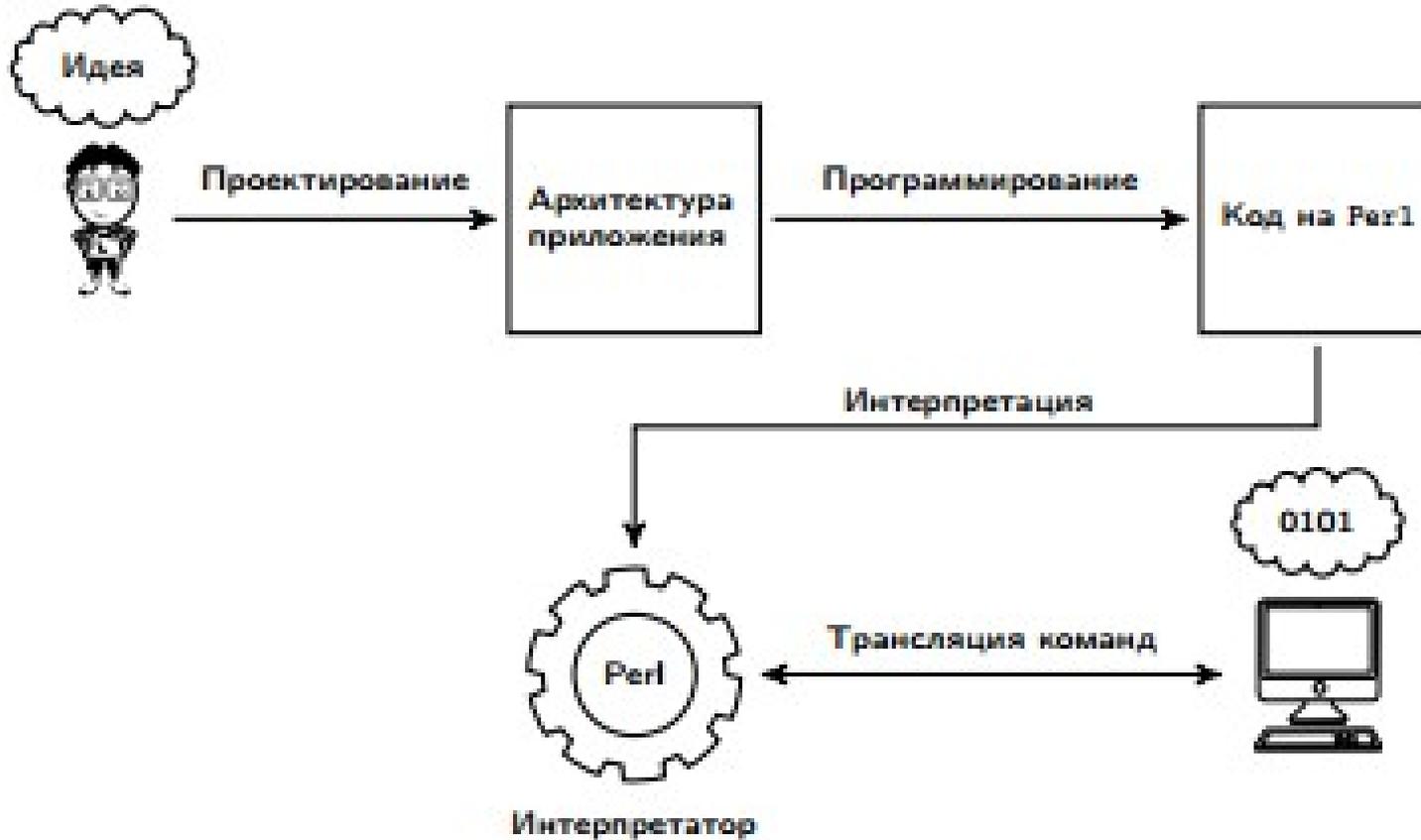
# Интерпретатор Shell



# КОМПИЛЯЦИЯ



# Интерпретация



Что такое программа ?

Скрипт(програ́мма) — последовательность инструкций, предназначенных для выполнения интерпретирующей машиной.

Имя переменной - это "контейнер" для хранения значения данных. Обращение к значению (получение значения) переменной называется подстановкой переменной.

# Присваивание значений и подстановка переменных

```
#!/bin/bash
```

```
a=375
```

```
hello=$a
```

# Присвоение через read

```
# разновидность присваивания:
```

```
echo -n "Введите \"a\" "
```

```
read a
```

```
echo "Значение \"a\" - $a."
```

# Особое присвоение

```
#!/bin/bash
```

```
a=23
```

```
echo $a
```

```
b=$a
```

```
echo $b
```

```
# Простой случай
```

```
# Теперь немножко фантазии (подстановка команд).
```

```
a=Áecho Hello!Á # Присваиваем результат  
выполнения ©echo© переменной ©a©
```

```
echo $a
```

# Задача на присвоение

#Даны две переменные:

$x=5$  ;

$y=3$  ;

Нужно поменять значение переменных местами.

# Проверка условия

```
test условие  
или  
[ условие ]
```

# Проверка прав доступа к файлу

- d Каталог
- a Обычный файл
- L Символическая связь
- r Файл для чтения
- s Файл имеет ненулевой размер
- w Файл для записей
- x Исполняемый файл

```
shell>[ -a file.txt ]
```

```
shell>echo $?
```

```
shell>test -w file.txt
```

# Применение логических операторов

-а Логическое AND , возвращает истину если обе части оператора принимают истинное значение.

-о Логическое OR, возвращает истину, если какая-либо из частей оператора может принимать истинное значение.

! Логическое NOT, возвращает истину, если условие ложно.

# Установлены ли для файлов права чтения

```
[ -w result.txt -a -w scores.txt ]
```

# Проверка строк

- = Две строки равны
- != Две строки не равны
- z Эта строка нулевая
- n Эта строка не является нулевой

Проверим , присвоено ли переменной нулевое значение

```
[ -z $EDITOR ]  
Это редактор vi ?  
[ $EDITOR = ©vi© ]
```

```
file1=©/dev/file.txt©  
file2=©/dev/file2.txt©  
[ "$file1" = "$file2" ] # равны ли  
названия фл.
```

# Проверка чисел

- eq Два числа равны
- ne Два числа не равны
- gt Первое число больше второго числа
- lt Первое число меньше второго
- le Первое число меньше или равно второму числу
- gt Первое число больше или равно второму числу.

```
$NUMBER=130
```

```
#[ "$NUMBER" -eq "130" ]
```

```
$echo $?
```

```
$source = 10
```

```
$dest = 20
```

```
#[ "$source" -gt "$dest" ]
```

```
#[ "120" -le "130" ]
```

# Конструкция ветвления if

```
if cmp a b &> /dev/null
then
    echo "Files are identical."
else
    echo "Files a and b differ."
fi
```

# Дополнительное условие

```
if [ condition1 ]  
then  
    command1  
    command2  
elif [ condition2 ]  
# Same as else if  
then  
    command4  
else  
    Default-command  
fi
```

```
echo "Проверяем 0
if [ 0 ]          # Ноль
then
    echo "0 - истина."
else
    # или ...
    echo "0 - ложь."
fi
```

```
echo "Проверяем 1
if [ 1 ]          # единица
then
    echo "1 - истина."
else
    echo "1 - ложь."
fi
```

## Задача.

Определить есть ли файл в системе и если его нет то создать с тем же названием.