

# Snakemake

Elementi di bioinformatica

# Regole snakemake

```
rule RULE_NAME:
    input:
        "INPUT_FILE"
    output:
        "OUTPUT_FILE"
    shell:
        '''
        Command to execute using input and output
        '''
```

# Regole snakemake - esempi

```
rule cat:  
    input:  
        "test.fa"  
    output:  
        "catted.fa"  
    shell:  
        '''  
        cat {input} > {output}  
        '''
```

# Regole snakemake - esempi

```
rule cat:  
    input:  
        f_in="test.fa"  
    output:  
        f_out="catted.fa"  
    shell:  
        '''  
        cat {input.f_in} > {output.f_out}  
        '''
```

# Regole snakemake - esempi

```
rule cat:
    input:
        f_in1="fa1.fa",
        f_in2="fa2.fa",
    output:
        f_out="catted.fa"
    shell:
        '''
            cat {input.f_in1} {input.f_in2} > {output.f_out}
        '''
```

# Regole snakemake - esempi

```
rule wc:  
    input:  
        "catted.fa"  
    output:  
        count=wc.txt  
    shell:  
        '''  
        wc {input} > {output.count}  
        '''
```

# Regole snakemake - esempi

```
rule grep:
    input:
        catted="catted.fa",
    output:
        s="seqs.txt",
        a="a.txt",
        c="c.txt",
    shell:
        '''
        grep "^>" {input.catted} -c > {output.g}
        grep "[Aa]" {input.catted} -c > {output.a}
        grep "[Cc]" {input.catted} -c > {output.c}
        '''
```

# Esecuzione

Salvare il file come “Snakefile”

- snakemake

Salvare in un file “test.smk”

- snakemake -s test.smk



# Esempio più realistico

```
rule index_fa:
    input:
        "hs37d5.fa"
    output:
        "hs37d5.fa.fai"
    shell:
        '''
        samtools faidx {input}
        '''
```

```
rule get_region:
    input:
        fa="hs37d5.fa",
        fai="hs37d5.fa.fai",
    output:
        "sub.fa"
    params:
        reg="21:34407410-40062514",
    shell:
        '''
        samtools faidx {input.fa} {params.reg} > {output}
        '''
```

# Esempio più realistico

```
rule index_fa:
    input:
        "hs37d5.fa"
    output:
        "hs37d5.fa.fai"
    shell:
        '''
        samtools faidx {input}
        '''
```

```
rule get_region:
    input:
        fa="hs37d5.fa",
        fai="hs37d5.fa.fai",
    output:
        "sub.fa"
    params:
        reg="21:34407410-40062514",
    shell:
        '''
        samtools faidx {input.fa} {params.reg} > {output}
        '''
```

# Esempio più realistico

```
rule index_fa:
    input:
        "hs37d5.fa"
    output:
        "hs37d5.fa.fai"
    shell:
        '''
        samtools faidx {input}
        '''
```

```
rule get_region:
    input:
        fa="hs37d5.fa",
        fai="hs37d5.fa.fai",
    output:
        "sub.fa"
    params:
        reg="21:34407410-40062514",
    shell:
        '''
        samtools faidx {input.fa} {params.reg} > {output}
        '''
```

snakemake -s test.smk -c1

# Esempio più realistico

```
rule index_fa:
    input:
        "hs37d5.fa"
    output:
        "hs37d5.fa.fai"
    shell:
        '''
        samtools faidx {input}
        '''
```

```
rule get_region:
    input:
        fa="hs37d5.fa",
        fai="hs37d5.fa.fai",
    output:
        "sub.fa"
    params:
        reg="21:34407410-40062514",
    shell:
        '''
        samtools faidx {input.fa} {params.reg} > {output}
        '''
```

```
snakemake -s test.smk -c1 -R get_region
```

# Regola *run* all'inizio

```
rule run:  
    input:  
        "sub.fa"
```

snakemake -s test.smk -c1

# DAG

snakemake -s test.smk --dag

```
digraph snakemake_dag {  
  graph[bgcolor=white, margin=0];  
  node[shape=box, style=rounded, fontname=sans, fontsize=10, penwidth=2];  
  edge[penwidth=2, color=grey];  
  0[label = "run", color = "0.22 0.6 0.85", style="rounded,dashed"];  
  1[label = "get_region", color = "0.44 0.6 0.85", style="rounded,dashed"];  
  2[label = "index_fa", color = "0.00 0.6 0.85", style="rounded,dashed"];  
  1 -> 0  
  2 -> 1  
}
```



# Expand

```
rule index_fa:
    input:
        "{sample}.fa"
    output:
        "{sample}.fa.fai"
    shell:
        '''
        samtools faidx {input}
        '''
```

# Expand

```
rule get_region:
    input:
        fa="{sample}.fa",
        fai="{sample}.fa.fai",
    output:
        "{sample}.sub.fa"
    params:
        reg="21:100-200",
    shell:
        '''
        samtools faidx {input.fa} {params.reg} > {output}
        '''
```



# Expand

```
rule run:
    input:
        expand(
            "{sample}.sub.fa.fai",
            sample=["a", "b", "c"]
        )
```

# Expand

```
rule run:
    input:
        expand(
            "{sample}.sub.fa.fai",
            sample=["a", "b", "c"]
        )
```

# Expand

```
rule run:
    input:
        expand(
            "{sample}.sub.fa.fai",
            sample=["a", "b", "c"]
        )
```

