

Title

Subtitle

Piotr Fulmański

`piotr@fulmanski.pl`

25 stycznia 2017

- Appears on all slides of the frame.
- Appears from the second slide onwards.
- Appears from the third slide onwards.

- Appears on all slides of the frame.
- Appears from the second slide onwards.
- Appears from the third slide onwards.

- Appears on all slides of the frame.
- Appears from the second slide onwards.
- Appears from the third slide onwards.

- Tylko 3
- Od 2
- **Od 1**
- Tylko 2 i 3
- **Zawsze**
- Tylko 4

- Tylko 3
- Od 2
- Od 1
- Tylko 2 i 3
- Zawsze
- Tylko 4

- Tylko 3
- Od 2
- Od 1
- Tylko 2 i 3
- Zawsze
- Tylko 4

- Tylko 3
- Od 2
- Od 1
- Tylko 2 i 3
- Zawsze
- Tylko 4

- Od 1
- Zawsze

- Od 2
- Od 1
- Tylko 2 i 3
- Zawsze

- Tylko 3
- Od 2
- Od 1
- Tylko 2 i 3
- Zawsze

- Od 2
- Od 1
- Zawsze
- Tylko 4

- First subject.
- Second subject.
- Third subject.

- First subject.
- Second subject.
- Third subject.

- First subject.
- Second subject.
- Third subject.

Tytuł bloku

Text w bloku

Tytuł bloku

Text w bloku

Tytuł bloku

Text w bloku


```
for i:=maxint to 0 do
begin
{
  do nothing
}
end;
Write('Case insensitive ');
Write('Pascal keywords.');
```

```
1 for i:=maxint to 0 do
  begin
3 { do nothing }
  end;
5 Write('Case_ insensitive_ ');
  Write('Pascal_ keywords.');
```

- ① 1
- ② 2
 - ① a
 - ② b
 - ③ c
- ③ 3

4 4

5 5

6 6

Algorithm 1 Pseudocode for the A* algorithm (part 1/2)

```
1: function findPathWithAStar
2:   open := new PriorityQueue
3:   closed := new Set
4:   open.add(S, eval(S), null)
```

Algorithm 2 Pseudocode for the A* algorithm (part 2/2)

```
5:   while (!open.isEmpty()) do
6:     n := open.get()                                ▷ Step 2.2
7:     closed.add(n)
8:     if n = F then
9:       return backtracked path
10:    end if
11:    for all child in n.getChildren() do
12:      if !(closed.has(child) || open.has(child)) then
13:        open.add(child, eval(child), n)
14:      else if open.has(child) then
15:        open.update(child, eval(child), n)
16:      end if
17:    end for
18:  end while
19:  return null                                       ▷ There is no path from S to F
20: end function
```

Tables and colors

	A	B	C	D	E
A		B	B	B	B
B	A		C	E	E
C	B	B		D	B
D	E	E	C		E
E	B	B	B	D	