پگاه گورکانی

ساختمان داده

تمرین سوم

سکشن دوشنبه

دکتر اسکندری

```
{
"nbformat": 4,
"nbformat_minor": 0,
"metadata": {
"colab": {
"provenance": []
},
"kernelspec": {
"name": "python3",
"display_name": "Python 3"
},
"language_info": {
"name": "python"
}
},
"cells": [
{
"cell_type": "code",
"execution_count": 132,
"metadata": {
"id": "Y87ET6Q-44Rc"
},
"outputs": [],
"source": [
"class Queue():\n",
"  def __init__(self, max_size):\n",
"    self.max_size = max_size\n",
"    self.Q = [0] * max_size\n",
"    self.num = 0\n",
"    self.first = 0\n",
"\n",
"  def enqueue(self, item):\n",
"    if self.num >= self.max_size:\n",
"      raise Exception(\"Queue overflow!\")\n",
"    self.Q[(self.num + self.first) % self.max_size] =
item\n",
"    self.num += 1\n",
"\n",
"  def dequeue(self):\n",
"    if self.num == 0:\n",
"      raise Exception(\"Queue empty!\")\n",
"    item = self.Q[self.first]\n",
"    self.first = (self.first + 1) % self.max_size\n",
"    self.num -= 1\n",
"    return item\n",
"\n",
"  def front(self):\n",
"    if self.num == 0:\n",
"      raise Exception(\"Queue empty\")\n",
"    return self.Q[self.first]\n",
"\n",
"  def is_empty(self):\n",
"    return self.num == 0\n",
"\n",
"  def size(self):\n",
"    return self.num\n",
"\n",
"  def is_full(self):\n",
"    return self.num >= self.max_size\n",
"\n",
"  def get_value_at_index(self, index):\n",
"    if self.num == 0:\n",
"      raise Exception(\"Empty Queue\")\n",
"    if self.num > self.max_size:\n",
"      raise Exception(\"Queue OverFlow\")\n",
"    return self.Q[(self.first + index) %
self.max_size]"
]
},
{
"cell_type": "code",
"source": [
"q = Queue(5)\n",
"\n",
"q.enqueue(1)\n",
"q.enqueue(2)\n",

"q.enqueue(3)\n",

"q.enqueue(4)\n",

"q.enqueue(5)\n",

"\n",

"q.dequeue()\n",

"\n",

"q.enqueue(6)"

],

"metadata": {

"id": "pQ3-HA04rvsn"

},

"execution_count": 133,

"outputs": []

},

{

"cell_type": "code",

"source": [

"print(q.get_value_at_index(0))"

],

"metadata": {

"colab": {

"base_uri": "https://localhost:8080/"

},

"id": "p-4aOavDt7PO",
"outputId":
"a1a3df10-f438-4958-c7a6-b3a8194d382f
"
},
"execution_count": 134,
"outputs": [
{
"output_type": "stream",
"name": "stdout",
"text": [
"2\n"
]
}
]
},
{
"cell_type": "code",
"source": [
"print(q.get_value_at_index(4))"
],
"metadata": {
"colab": {
"base_uri": "https://localhost:8080/"
},
"id": "pVKoXUKBt9VE",
"outputId":
"9b3140b5-74c1-4fc6-bb50-
```

```
6048f71f4730"
},
"execution_count": 135,
"outputs": [
{
"output_type": "stream",
"name": "stdout",
"text": [
"6\n"
]
}
]
}
]
}
```